Institutional Repository - Research Portal
Dépôt Institutionnel - Portail de la Recherche
researchportal.unamur.be
University of Namur

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

Analyse technique des formats MPEG

Brasseur, Christophe; Tixhon, Rudy

Award date:
2002

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 05. May. 2023

MEMOIRE DE FIN D'ETUDES DE
CHRISTOPHE BRASSEUR ET RUDY TIXHON

« Analyse Technique des formats MPEG »

ANNEE ACADEMIQUE 2001-2002

FACULTES UNIVERSITAIRES NOTRE-DAME DE LA PAIX, NAMUR
INSTITUT D'INFOMATIQUE
ANNEE ACADEMIQUE 2001-2002

Analyse Technique

des formats

MPEG

par Christophe Brasseur
et Rudy Tixhon

MEMOIRE PRESENTE EN VUE DE L'OBTENTION DU GRADE DE
MAITRE ET/OU LICENCIE EN INFORMATIQUE

LBS 10074818

RESUME

Ce travail fut réalisé dans le cadre du mémoire de fin d'étude de M. Christophe Brasseur et M. Rudy Tixhon sous l'encadrement de M. Jean-Marie Jacquet, professeur à l'institut d'informatique de l'université de Namur.

Vu l'émergence des nouvelles technologies ces dernières années dans le domaine de l'imagerie numérique, l'étude des travaux existants dans ce domaine semble être une opportunité intéressante. Dès lors, il nous est paru pertinent d'orienter notre travail plus particulièrement vers le groupe MPEG, institution reconnue mondialement, et en particulier vers les différentes normes qu'ils ont élaborées à ce jour.

Les normes abordées dans ce document sont le MPEG-1, MPEG-2, MPEG-4 et MPEG-7, la norme MPEG-21 étant encore trop jeune pour pouvoir être analysée en détail.

ABSTRACT

This present document has been made in the context of the report of M. Christophe Brasseur and M. Rudy Tixhon for the last year of their Mastering in Computer Sciences and was supervised by M. Jean-Marie Jacquet, Professor at the Institute of Computer Science of the University of Namur.

The aim of this document is to realise a technical analysis of the different formats developed by the worldwide known MPEG group. This choice seems to be really pertinent because of the current emergence of new technologies in the digital imagery domain nowadays.

The MPEG-1, MPEG-2, MPEG-4 and MPEG-7 standards are described in our document, the last standard in date, MPEG-21 was still in '*work in progress*' so we decided to ignore it.

AVANT-PROPOS

Nous tenons à remercier Monsieur Jean-Marie Jacquet
pour son temps et ses précieux conseils
qui nous ont permis de réaliser ce mémoire
et également à Christian pour le temps
qu'il a consacré à corriger nos nombreuses
fautes d'orthographe …

TABLE DES MATIERES

FIGURES

TABLEAUX

PARTIE 1

PREFACE

I - PREFACE

De tous temps l'homme a été conduit à inventer des techniques lui permettant de repousser les limites du temps et de l'espace. Des formes les plus anciennes de communication visuelle telles que le dessin, la peinture et l'écriture, on est passé à la photographie, au téléphone, à la télévision, et plus récemment aux jeux vidéo et au World Wide Web.

Au début l'Internet fut utilisé comme un vecteur de communication de messages textuels. Bien vite l'appétit de conquête de l'homme l'a poussé à imaginer d'autres applications dépassant ces limites textuelles et permettant l'accès aux images et aux sons (multimédia).

Mais la digitalisation du son et des images a un coût élevé en terme de masse de données. Prenons un exemple : quelle peut être la taille d'un fichier contenant une heure de film ?

Prenons le cas d'un moniteur dont la résolution est de 1024 x 768. Un tel écran représente une matrice de pixels de 1024 x 768, on a donc 786.432 pixels.

Il faut 32 bits, soit 4 octets, pour coder la couleur d'un pixel (ce codage permet de représenter l'ensemble des couleurs possibles), ce qui donne 3.145.278 (ou 3072 Ko). Pour obtenir une seconde de film, il faut encoder 25 images par seconde, on a donc 3072 x 25 donnant 76800 Ko (ou 75 Mo/s). En résumé un espace disque de 263 Go est nécessaire pour stocker une heure de film.

D'où se sont posés les problèmes du stockage des contenus multimédias numérisés et de leur transmission au travers de réseaux à capacité limitée en terme de bande passante.

Les premières solutions envisagées ont été de diminuer la palette de couleurs, de baisser la résolution, de baisser le framerate, etc. Mais ces solutions dégradent fortement la qualité de l'image et le gain en terme de place n'est pas suffisant.

Dès lors, il est apparu indispensable de développer des outils de compression des données numériques multimédias permettant de diminuer la quantité de données à transmettre et de limiter les espaces mémoires nécessaires. MPEG-1 fut destiné au stockage des contenus numériques, MPEG-2 a été crée en vue de permettre la télévision numérique, et le MPEG-4 pour le transport d'informations numériques sur des canaux à faibles débits.

Un autre problème se pose par ailleurs : ayant la possibilité d'échanger des contenus multimédias à travers des réseaux de transmission, il faut pouvoir localiser ces contenus. C'est là qu'intervient la norme MPEG-7 qui a pour but d'attacher une sémantique à ces contenus afin de satisfaire le consommateur dans sa quête du Graal audiovisuel (le bon contenu au bon moment).

Une des caractéristiques remarquable de ces développements est la convergence des techniques. En effet, très récemment encore, les technologies utilisées pour transmettre le son et l'image avaient ceci de particulier qu'elles avaient très peu de choses en commun. Le son

gravé sur disque en vinyle par exemple utilisait des principes très différents de ceux de la cinématographie ou du disque laser.

Concernant des aspects plus pragmatiques, quelques mots sur la répartition du travail nécessaire à la création de ce mémoire. Rudy s'est plus particulièrement penché sur les normes MEG-1 et MPEG-4 tandis que Christophe s'est davantage attaché aux normes MPEG-2 et MPEG-7.

Partie 2

Introduction

au MPEG et a la Normalisation

II - INTRODUCTION AU MPEG ET A LA NORMALISATION

II.1 - LE GROUPE MPEG

Le *Moving Picture Coding Expert Group* (*MPEG*) est né en 1988 dans le cadre du JTC1, Comité technique conjoint ISO/IEC sur la technologie de l'information, avec pour mission le développement de normes pour la représentation codée des images et du son en vue de leur enregistrement et de leur extraction sur DSM (***Digital Storage Media***). Il devint le Groupe de travail 11 (*WG 11*) du JTC 1/SC 29, en novembre 1991.

Le « nom de code » complet attribué au MPEG est ISO/IEC JTC1/SC29/WG11

Lors de sa première réunion en mai 1988, 25 experts étaient présents, maintenant, chaque année se tiennent généralement trois réunions du comité MPEG rassemblant plus de 300 experts de plus de 20 pays différents.

Le souci du groupe MPEG était de pouvoir établir des standards permettant à différents pays de pouvoir utiliser les mêmes médias sans rencontrer des problèmes d'incompatibilité. Comme ce fut le cas pour les différents formats de télévision par exemple.

Le groupe MPEG s'est constitué dans un double but : d'abord trouver une méthode pour convaincre les industries de l'avantage technologique d'une solution commune pour passer ensemble au numérique ; ensuite et surtout, définir une *syntaxe* unique capable de représenter l'information audiovisuelle et de devenir la plate-forme commune qui permettra l'interopérabilité entre les applications.

Si au début le MPEG ne devait standardiser que la vidéo, il parut rapidement évident que le son devait être également traité pour fournir aux utilisateurs une solution intégrée.

La normalisation était, avant MPEG, un processus lent, survenant dans de nombreux cas a posteriori, c'est-à-dire qu'elle avalisait une solution déjà adoptée par le marché. Au lieu de cela, l'objectif de MPEG fut de faire intervenir la normalisation à priori, anticipant les besoins du marché avant que les industries ne soient engagées trop en avant dans d'importants investissements.

En particulier le MPEG-1 est la première norme ayant vu le jour en 1988 sous l'appellation ISO/IEC-11172. MPEG-1, la mission première de MPEG était le développement de normes pour la représentation codée des images et du son en vue de leur enregistrement et de leur extraction sur DSM (***Digital Store Media***) à un débit de 1,5 Mbps.
Son application la plus connue est le VCD (Video CD), ayant surtout eu du succès sur le continent asiatique comme nous le verrons plus tard.

La norme MPEG-2 (ISO/IEC-13818), débutée en 1990, est essentiellement issue de MPEG-1 avec quelques modifications pour tenir compte de l'application phare qui était la télévision

numérique, et sur laquelle est notamment basé le DVD (***Digital Versatile Disc***). Les principales composantes de MPEG-2 ont été terminées fin 1994.

La norme MPEG-3 devait spécifier le codage d'un signal haute définition (20-40 Mbps). Mais, rapidement, ce pré requis a été pris en compte dans la définition de la norme MPEG-2 sans changement significatif des méthodes de codage.

En 1998 ce fut le tour du MPEG-4 (ISO/IEC-14496), introduisant une philosophie très différente de ses prédécesseurs basée sur la possibilité d'isoler des objets audio-visuels au sein d'une séquence vidéo.

La norme MPEG-7 (ISO/IEC-15938) marque un changement radical car cette norme spécifie la façon dont les données sont représentées et non plus la manière dont elles sont compressées. Le développement a commencé en octobre 1996. Le MPEG-21 (ISO/IEC-21000) quant à lui, est encore en développement et a pour objectif de gérer tout le coté copyright de la vidéo.

Vu que les standards émis par MPEG ont une importance stratégique élevée pour nombre d'entreprises à travers le monde, il n'est pas étonnant que ces décisions soit réglementées par des directives émanant de l'ISO, de l'IEC et du JTC1.

Voici une présentation générale de l'organisation internationale de normalisation (ISO)
Ci-dessous sont détaillés les organes de normalisation les plus importants qui interviennent dans le processus de création des normes MPEG, à savoir l'ISO, l'IEC et le groupe de travail MPEG à proprement parlé.

II.2 - LA NORMALISATION INTERNATIONALE

II.2.1 - Définition d'une norme

« Les normes sont des accords documentés contenant des spécifications techniques ou autres critères précis destinés à être utilisés systématiquement en tant que règles, lignes directrices ou définitions de caractéristiques pour assurer que des matériaux, produits, processus et services sont aptes à leur emploi. »

II.2.2 - Différence entre norme et standard

En anglais, un seul terme existe : *Standard* mais en français, deux termes se côtoient : le **standard** et la **norme.**
On a coutume, d'opposer les **normes**, documents validés par des instances officielles et qui, de ce fait même offrent une certaine garantie de stabilité et de pérennité, aux **standards**, états de fait résultant de mécanismes économiques et traduisant souvent la domination d'un industriel ou d'un groupe d'industriels sur un marché.

L'on peut par exemple considérer Microsoft Windows comme un standard lors de l'achat d'un nouvel ordinateur, il est en effet particulièrement difficile d'acheter un PC sans système d'exploitation ou avec Linux installé.

II.2.3 - Historique

La normalisation internationale commença dans le domaine électrotechnique avec la création, en 1906, de la Commission électrotechnique internationale (CEI). Les premiers travaux fondamentaux dans d'autres domaines furent entrepris par la Fédération internationale des associations nationales de normalisation (ISA), créée en 1926. Au sein de l'ISA, l'accent portait de façon prépondérante sur l'ingénierie mécanique.
Les activités de l'ISA cessèrent en 1942 en raison de la Seconde Guerre mondiale. À la suite d'une réunion tenue à Londres en 1946, les délégués de 25 pays décidèrent de créer une « *nouvelle organisation internationale dont l'objet serait de faciliter la coordination et l'unification internationales des normes industrielles* ». La nouvelle organisation non gouvernementale, ISO, entra officiellement en fonction le 23 février 1947.
La première norme ISO fut publiée en 1951 sous le titre « *Température normale de référence des mesures industrielles de longueur* ».

II.2.4 - L'ISO

II.2.4.1 - Qu'est ce que l'ISO ?

L'organisation internationale de normalisation (ISO) est une fédération mondiale d'organismes nationaux de normalisation de quelques 140 pays, à raison d'un organisme par pays.
L'ISO a pour mission de favoriser le développement de la normalisation et des activités connexes dans le monde, en vue de faciliter entre les nations les échanges de biens et de services et de développer la coopération dans les domaines intellectuels, scientifiques, techniques et économiques.
Les travaux de l'ISO aboutissent à des accords internationaux qui sont publiés sous la forme de Normes internationales.

Il est à noter que 'ISO' n'est pas un sigle. C'est un nom qui a été choisi en pensant au grec *isos* qui signifie 'égal', probablement parce que tous les industriels et les consommateurs sont égaux face à une norme.
L'ISO a trois langues officielles : l'anglais, le français et le russe ; en anglais, on l'appelle *International Organization for Standardization* et non *International Standards Organization*.

II.2.4.2 - Pourquoi la normalisation internationale est-elle nécessaire?

La normalisation internationale a pour but d'éviter l'existence de normes non harmonisées pour des technologies semblables, dans des pays ou des régions différents. Les industries tournées vers l'exportation ont depuis longtemps senti la nécessité de s'accorder sur des normes mondiales pour aider à rationaliser le processus des échanges internationaux. C'est cet objectif, justement, qui a présidé à la création de l'ISO.

Les domaines d'activité concernés sont nombreux et divers tels que le traitement de l'information et les communications, le textile, l'emballage, la distribution des marchandises, la production et l'utilisation de l'énergie, la construction navale, les services bancaires et financiers.

Les utilisateurs accordent une plus grande confiance aux produits et services qui sont conformes à des Normes internationales. L'assurance de cette conformité peut être fournie par le biais d'une déclaration du fabricant ou par des audits effectués par des organismes indépendants.

En effet par exemple, jusqu'en 1989, le marché du CD-Rom a stagné. Comme il n'y avait pas d'accord entre les différents industriels sur le formatage des données stockées, les éditeurs étaient peu enclins à adopter le CD-Rom comme support de leurs publications, en raison du risque de dépendance technologique. Lorsque les constructeurs se sont accordés sur une norme internationale, ce risque de dépendance a disparu, les CD-Rom devenant lisibles sur une gamme plus large de matériels.

II.2.4.3 - Les réalisations de l'ISO

Voici quelques exemples de normes ISO :

- La ***désignation ISO de la sensibilité des pellicules***, parmi bien d'autres normes concernant le matériel photographique, a été adoptée mondialement, facilitant singulièrement les choses pour l'utilisateur.
- Grâce à la normalisation du format des ***cartes de téléphones et des cartes bancaires***, celles-ci peuvent être utilisées dans le monde entier.

Des dizaines de milliers d'entreprises mettent en oeuvre la série ***ISO 9000*** qui fournit un cadre pour le management et l'assurance de la qualité. La série ***ISO 14000*** fournit un cadre similaire pour le management environnemental.

- ***Les Formats de papier***.
- ***Les codes internationaux ISO pour les noms de pays, les monnaies et les langues***.

Le champ d'action de l'ISO ne se limite pas à un secteur particulier. Il couvre tous les domaines techniques, à l'exception de l'ingénierie électrique et électronique, qui sont du ressort de la CEI. Les travaux dans le domaine des technologies de l'information sont menés par un comité technique mixte ISO/CEI, pour une liste exhaustive des secteurs d'activité, veuillez consulter la page web :
http://www.iso.ch/iso/fr/aboutiso/introduction/TechnicalCommitteeList.TechnicalCommitteeList

II.2.4.4 - Qui sont les membres de l'ISO ?

L'ISO est composée de membres qui sont répartis en trois catégories :

Les **comités membres** de l'ISO sont les organismes nationaux les plus représentatifs de la normalisation dans leurs pays. Il en découle qu'un seul organisme par pays peut être admis en qualité de membre de l'ISO.

> Les comités membres ont le droit de participer et d'exercer leur droit de vote complet au sein des comités techniques et comités chargés de l'élaboration d'orientations politiques de l'ISO.

Un **membre correspondant** est en général une organisation dans un pays qui n'a pas encore entièrement développé son activité nationale en matière de normalisation.

> Les membres correspondants ne prennent pas une part active aux travaux techniques et d'élaboration des politiques, mais ont le droit d'être tenus pleinement informés des travaux qui présentent pour eux un intérêt.

L'ISO a créé aussi une troisième catégorie de membres, **membre abonné**, pour des pays à économie très limitée.

> Ces membres abonnés paient une cotisation réduite qui leur permet néanmoins de rester en contact avec la normalisation internationale.

II.2.4.5 - Qui fait le travail ?

Les travaux techniques de l'ISO sont menés au sein d'une structure hiérarchisée comptant quelques 2850 comités techniques, sous-comités et groupes de travail. Dans le cadre de ces comités, des représentants qualifiés des milieux industriels, des instituts de recherche, des autorités gouvernementales, des organismes de consommateurs et des organisations internationales du monde entier se retrouvent en partenaires à droits égaux dans la recherche de solutions à des problèmes de normalisation d'envergure mondiale.

Quelques 30 000 experts participent aux réunions chaque année.

La responsabilité principale de l'administration d'un comité de normalisation est assumée par l'un des organismes nationaux de normalisation qui forment l'ISO : AFNOR, ANSI, BSI, CSBTS, DIN, SIS, etc.

Le Secrétariat central à Genève a pour rôle d'assurer une circulation fluide de la documentation dans toutes les directions, de clarifier les questions d'ordre technique avec les secrétariats et les présidents et d'assurer la mise au point rédactionnelle et l'impression des accords approuvés par les comités techniques, ainsi que leur soumission, en tant que projets de Normes internationales, au vote des comités membres de l'ISO et, enfin, leur publication. Les réunions des comités techniques et des sous-comités sont convoquées par le Secrétariat central, qui coordonne l'ensemble de ces réunions avec les secrétariats des comités avant d'en fixer la date et le lieu.

Tout comité membre qui s'y intéresse a le droit d'être représenté au sein du comité traitant d'un sujet particulier. Les organisations internationales gouvernementales et non gouvernementales ayant des liaisons avec l'ISO prennent également part aux travaux. L'ISO collabore étroitement avec la Commission électrotechnique internationale (CEI) sur toutes les questions de normalisation électrotechnique.

II.2.4.6 - Comment élabore-t-on une norme ISO ?

L'élaboration d'une norme ISO fait appel aux principes suivants :

- **Consensus**
  Les points de vue de tous les intéressés sont pris en compte : fabricants, vendeurs et utilisateurs, groupes de consommateurs, laboratoires d'essais, gouvernements, professionnels de l'ingénierie et organismes de recherche.

- **À l'échelle de l'industrie**
  Solutions globales visant à satisfaire les industries et les clients partout dans le monde.

- **Volontaire**
  La normalisation internationale étant mue par le marché, elle s'appuie sur la participation volontaire de tous les protagonistes du marché.

II.2.4.6.1 - Les phases de l'élaboration d'une norme ISO

Le processus d'élaboration des normes ISO comporte trois phases principales :

- *PHASE 1* : Le besoin d'une norme est en général manifesté par un secteur de l'industrie, qui fait part de ce besoin à un comité membre national. Ce dernier soumet le projet à l'ISO dans son ensemble. Lorsque le besoin d'une Norme internationale a été reconnu et formellement approuvé, la première phase consiste à **définir l'objet technique de la future norme**. Cette phase se déroule normalement au sein de groupes de travail constitués d'experts provenant des pays intéressés par la question.

- *PHASE 2* : Lorsqu'un accord est atteint sur les aspects techniques devant faire l'objet de la norme, une deuxième phase commence au cours de laquelle les pays négocient les détails des spécifications qui devront figurer dans la norme. Il s'agit de la **phase de recherche de consensus**.

- *PHASE 3* : La dernière phase comprend **l'approbation formelle du projet** de Norme internationale (les critères d'acceptation stipulent que le document doit être approuvé par les deux tiers des membres de l'ISO qui ont participé activement au processus d'élaboration de la norme et par 75 % de l'ensemble des membres votant), à la suite de quoi le texte est publié en tant que Norme internationale ISO.

Vous pouvez trouver de plus amples détails sur l'élaboration des normes dans l'ANNEXE.

La plupart des normes doivent être revues périodiquement. Plusieurs facteurs concourent à faire en sorte qu'une norme soit dépassée: évolution des techniques, méthodes nouvelles et nouveaux matériaux, exigences nouvelles en matière de qualité et de sécurité. Pour tenir compte de ces facteurs, l'ISO s'est fixé pour règle générale que toutes les normes ISO doivent être revues à des intervalles n'excédant pas cinq ans. Il est nécessaire, parfois, de réviser une norme à plus brève échéance.

À ce jour, les travaux de l'ISO ont abouti à la publication de quelques 13 000 Normes internationales, représentant plus de 400.000 pages en anglais et en français (les normes terminologiques comprenant souvent d'autres langues).

II.2.4.7 - Comment les travaux de l'ISO sont-ils financés ?

Le financement de l'ISO traduit fidèlement son mode de fonctionnement décentralisé avec d'une part le financement des activités du Secrétariat central et d'autre part, le financement des travaux techniques proprement dits.

Le financement du Secrétariat central provient des cotisations des membres (80 %) et des recettes de la vente des normes et autres publications de l'Organisation (20 %).

Les comités membres de l'ISO assument les dépenses nécessaires au fonctionnement de chacun des secrétariats techniques dont ils ont la charge.

À cela il faut encore ajouter la valeur de l'apport volontaire de quelque 30.000 experts en termes de temps et de voyages.

II.2.4.8 - JTC, SC et WG

Pour ce qui est plus particulièrement des technologies de l'information, l'ISO a décidé en 1987, pour regrouper des efforts autrefois dispersés, de former la CEI un comité technique mixte, le JTC1 (***J**oint **T**echnical **C**ommitee **1***) ou ISO/CEI JTC1.
C'est ce qui explique que, souvent, on se réfère à une norme du JTC1 en écrivant ISO/CEI JTC1 suivi d'un numéro.

Le JTC1 est, de très loin, le plus gros des 185 comités techniques de l'ISO.
Le JTC1 est responsable de près de 1500 normes ou DIS. Il regroupe 26 pays participants soit pratiquement tous les grands pays développés et 36 pays observateurs. Il tient une réunion plénière tous les 9 mois. On estime qu'environ 2200 experts du monde entier participent aux travaux JTC1.

Les sujets d'études peuvent passer d'un sous-comité à un autre. C'est ainsi que si les travaux sur la compression des images ont commencé dans le SC2, la création du SC29 a fait qu'ils se sont poursuivis en son sein.
Par ailleurs, chaque sous-comité est divisé en un certain nombre de groupes de travail (WG)

Par exemple le WG 11 fait partie du SC 29 et est destiné au codage de l'image animée et du son c'est-à-dire le MPEG.

II.2.4.9 - Les partenaires

- *Les partenaires internationaux*

  L'ISO collabore avec son homologue de la normalisation internationale, la CEI, dont le domaine d'activité complète le sien. Ensemble, l'ISO et la CEI coopèrent avec l'UIT (***U**nion **I**nternationale des **T**élécommunications*). Comme l'ISO, la CEI est un organisme non gouvernemental, alors que les membres de l'UIT, agence spécialisée de l'Organisation des Nations Unies, sont des gouvernements. Les trois organisations collaborent étroitement dans la normalisation des technologies de l'information et des télécommunications.

  L'ISO édifie un partenariat stratégique avec l'Organisation mondiale du commerce (*OMC*), l'objectif commun étant de promouvoir un système mondial de libre-échange

équitable. Les accords politiques obtenus dans le cadre de l'OMC doivent s'appuyer sur des accords techniques.

- Partenaires régionaux

De nombreux membres de l'ISO sont également membres d'organisations régionales de normalisation. Cette situation permet à l'ISO de créer plus facilement des ponts avec les activités régionales de normalisation dans le monde. L'ISO a reconnu des organisations régionales de normalisation représentant l'Afrique, les pays arabes, la région couverte par la Communauté des États indépendants, l'Europe, l'Amérique latine, la zone Pacifique et les nations de l'Asie du Sud-Est. Cette reconnaissance se fonde sur l'engagement pris par les organismes régionaux d'adopter les normes ISO – sans modification, chaque fois que possible – comme normes nationales de leurs pays membres et de ne procéder à l'élaboration de normes divergentes que s'il n'existe aucune norme ISO susceptible d'être adoptée directement.

L'ISO est aussi en liaison avec quelques 500 organisations internationales et régionales intéressées à certains aspects spécifiques de ses travaux de normalisation.

PARTIE 3

LE MPEG-1

III - MPEG-1 (ISO/IEC-11172)

Une norme MPEG spécifie deux points essentiels : la *structure du flux* et la *méthode de décodage* pour restituer le signal audiovisuel. La sélection des modes de codage mis en œuvre est laissée à l'utilisateur qui peut donc faire son propre compromis entre performance de l'équipement de compression et la complexité (et donc coût) d'implémentation.

Commençons ce premier chapitre par la découverte de la première norme du groupe MPEG c'est-à-dire le MPEG-1. Pour cela, présentons tout d'abord les techniques fondamentales sur lesquelles repose cette norme.

III.1 - ANALYSE TECHNIQUE MPEG-1

III.1.1 - MPEG-1 Partie 2: Vidéo (ISO/IEC 11172-2)

Souvent, les séquences vidéo possèdent une redondance élevée au sein de leurs images ou entre leurs images. Le MPEG va utiliser ces redondances pour pouvoir réduire la bande passante nécessaire pour transmettre ou stocker ces séquences.
Cela se traduira généralement par un processus de compression des images originelles.
Il faudra continuellement chercher à trouver le meilleur équilibre entre la place occupée et la qualité obtenue.

En effet deux types de compressions sont envisageables, une sans perte de qualité (*lossless compression*) et une avec perte (*lossy compression*).

- Pour le **lossless** le but est d'obtenir une image compressée identique à son image modèle mais plus légère.
- En **lossy** l'objectif est d'atteindre un bit rate donné, le degré de qualité pouvant varier selon un critère objectif ou subjectif.

En règle générale, une image sera plus facilement compressible si elle contient beaucoup de redondances et cela dépendra bien évidemment de la complexité de la technique de compression.

Dans un signal vidéo, les redondances sont de deux types

III.1.1.1 - Types de redondances

- *Redondances spatiales*
  Les valeurs des pixels dans une même image ne sont pas indépendantes. On mettra donc à profit cette corrélation par l'utilisation d'une transformée orthogonale puis codage des coefficients pour compresser le signal. Ce mode de codage est appelé **Intra** *(ce mode sera vu plus en détail dans le chapitre intitulé : Type de Codage 'Intra')*. On notera que cette corrélation est variable en fonction du contenu de

l'image : des sources vidéo avec beaucoup de détails visuels contiennent assez peu de redondances spatiales. La compression du signal mettant à profit ce type de redondance étant dans ce cas peu efficace.

- *Redondances temporelles*
  Le même type de corrélation existe entre les pixels de deux images successives. On peut donc prédire le contenu d'une image par référence à une image précédente ou suivante et proche d'un point de vue contenu. Il suffit ensuite de coder le signal résiduel, qui représente les changements entre les deux images, de la même façon que pour les redondances spatiales. Ce mode de codage est appelé **Inter** *(ce mode sera vu plus en détail dans le chapitre intitulé : Type de Codage 'Inter')*

  Encore une fois, l'efficacité de cette compression du signal dépend du contenu. Une séquence vidéo contenant des mouvements peu modélisables ou de nombreux changements de scène favorisera assez peu la compression par réduction des redondances temporelles et sera l'objet d'une dégradation plus importante du signal pour un même débit de transmission.

III.1.1.2 - La vue chez l'être humain

Les techniques de compression utilisées sont en grande partie basées sur la connaissance que nous avons de la façon dont l'œil et le cerveau humain reconnaissent les images.
Lorsque qu'un humain « regarde », d'abord il distingue les détails d'une scène (il perçoit la résolution spatiale), puis il reconnaît les changements dans la scène (il perçoit la résolution temporelle).
Le processus de la vision se caractérise par le fait que les objets reflètent la lumière qui entre dans l'œil et stimule les photorécepteurs de la rétine. L'image qui apparaît est ensuite traitée par le cerveau.
Il existe deux types de photorécepteurs : les bâtonnets et les cônes. Les bâtonnets sont capables de distinguer le blanc et le noir ; les cônes permettent de distinguer les couleurs. Il y a plusieurs types de cônes, qui sont spécialement sensibles au rouge, au vert ou au bleu. Une caractéristique importante est le nombre et la distribution des bâtonnets et des cônes sur la rétine. Au centre de la rétine par exemple, on ne trouve que des cônes. C'est pour cela que l'oeil est relativement moins sensible à la couleur, et plus spécifiquement aux changements de couleur. Les techniques de compression vidéo utilisent ce manque de sensibilité aux couleurs en réduisant les informations concernant la couleur par image.

III.1.1.3 - Type de Codage 'Intra'

III.1.1.3.1 - Les espaces de couleur

Un espace de couleur est un modèle théorique décrivant comment séparer une couleur en composants. Il définit également la signification de ces composants. Il existe deux types d'espaces de couleur utilisés dans le domaine de la vidéo digitale : RGB et YUV.

- *RGB*
  RGB est utilisé dans le domaine des ordinateurs.
  Dans l'espace de couleur RGB, chaque pixel de l'écran a une valeur RGB correspondante. Une valeur RGB est construite sur base de trois composants, qui définissent une valeur pour le rouge, une valeur pour le vert et une valeur pour le bleu.

Dans le monde des ordinateurs, un certain nombre de bits est assigné pour chaque pixel. Afin de reproduire toutes les couleurs que l'être humain peut distinguer, chaque composant RGB doit être décrit sur 8 bits (2^{24} = 16.000.000 de couleurs différentes).
Cependant, RGB présente deux inconvénients majeurs. Premièrement, il faut le même nombre de bits pour chacun de ses composants afin de créer toutes les couleurs possibles. Deuxièmement, RGB n'est pas adapté pour les applications de télévision. Si on voulait augmenter la luminosité d'une image avec une télécommande RGB, cela entraînerait une augmentation de la valeur de chaque composant RGB.

- *YUV*

YUV est utilisé dans les téléviseurs standard.
La couleur est déterminée par un composant d'intensité (Y) et deux composants de couleur (U et V). Y, qui représente les informations de blanc et de noir, est appelé luminance, tandis que U et V sont appelés chrominance.
Pour les applications TV, cet espace de couleur a quelques avantages. Premièrement, il est facile de supporter les images en noir et blanc. Dans ce cas, seules les informations de luminance seront utilisées. Deuxièmement, l'espace de couleur YUV correspond mieux à notre perception de la couleur.
Les techniques de compression MPEG se basent sur le modèle YUV.

Il est à noter que des formules permettent de passer des valeurs YUV aux valeurs RGB.

$$Y = +0.299 * R + 1.587 * G + 0.114 * B$$
$$U = -0.299 * R - 0.587 * G + 0.886 * B$$
$$V = +0.701 * R - 0.587 * G - 0.114 * B$$

Pour certaines applications, les informations de couleur pour chaque pixel peuvent être moins précises que les informations de luminance. Dans ce cas, il est possible d'assigner des valeurs de chrominance seulement tous les deux pixels.
Plus formellement, on a défini plusieurs ratios d'échantillonnage. Le premier est 4:4:4 dans ce cas, les informations de luminance et de chrominance sont présentes pour chaque pixel.
Le deuxième est 4:2:2 dans ce cas les informations de luminance sont présentes pour chaque pixel tandis que les informations de chrominance sont présentes tous les deux pixels (dans la direction horizontale).
Les deux derniers sont 4:2:0 et 4:1:1. L'information de chrominance est présente tous les 4 pixels.

Tableau III-1 : Les différents formats de codage pour une image type de taille 720 par 480

Format A :B :C	Nombre de colonnes Y	Nombre de lignes Y	Nombre de colonnes C	Nombre de lignes C	Facteur de division horizontale	Facteur de division verticale
4 :4 :4	720	480	720	480	Aucun	Aucun
4 :2 :2	720	480	360	480	2 :1	Aucun
4 :2 :0	720	480	360	240	2 :1	2 :1
4 :1 :1	720	480	180	480	4 :1	Aucun
4 :1 :0	720	480	180	120	4 :1	4 :1

Par exemple, pour un codage 4:2:2, la luminosité sera codée sur 720 par 480, elle aura donc la même taille que l'image originale. Mais les chrominances seront codées sur 360 par 480, on aura donc appliqué un facteur de réduction de 2:1 sur l'horizontale. Suivant le type de codage utilisé il faut donc moins de place pour coder les chrominances.

Lors du développement de la télévision et de l'informatique, plusieurs organismes se sont formés pour essayer de standardiser les caractéristiques des équipements vidéo. Il existe donc plusieurs résolutions d'image

Tableau III-2 : Normes pour la résolutions d'une image

Normes	Organisme	Résolution en Pixels	Commentaires
NTSC	NTSC	640 x 480	format 4:3 défini en 1953, le nombre de lignes est en fait de 525 mais environ 8% de la bande est utilisée pour la synchronisation des équipements.
PAL		768 x 576	format 4:3 défini en 1963, 625 lignes réelles
SECAM		768 x 576	format 4:3 défini en 1959, 625 lignes réelles
HDTV	SMPTE	1920 x 1035	format 16:9 , 1125 lignes réelles
SIF NTSC	ISO	352 x 240	Utilisé par MPEG-1
SIF PAL/SECAM	ISO	352 x 288	Utilisé par MPEG-1
CCIR 601 525/60	ITU-R	720 x 480	pour NTSC, 858 x 525 réelles. Le nombre varie entre 704 et 720 et entre 480 et 496
CCIR 601 625/50	ITU-R	720 x 576	pour PAL / SECAM Le nombre varie entre 704 et 720
SQCIF	ITU-T	128 x 96	Sub-QCIF
QCIF	ITU-T	176 x 144	Quarter-CIF, défini pour les formats basse résolution
CIF	ITU-T	352 x 288	Common Intermediate Format, défini pour les formats basse résolution, utilisé par H-261 et H-263

4 CIF	ITU-T	704 x 576	Pour H-263
16 CIF	ITU-T	1408 x 1152	Pour H-263

Les formats classiquement utilisés en compression vidéo sont :

SIF (352 x 288 x 25 Hz 1:1 ou 352 x 240 x 30 Hz 1:1)
TV (720 x 578 x 50 Hz 2:1 ou 720 x 480 x 60 Hz 2:1)
HDTV (1440 x 1150 x 50 Hz 2:1 ou 1920 x 1080 x 60 Hz 2:1)

Avant le processus de compression lui-même, plusieurs modifications du format de la source peuvent s'avérer souhaitables ou même nécessaires.

III.1.1.3.2 - Conversion 4:2:2/4:2:0

La plupart des signaux vidéo générés par les cameras ont un échantillonnage de la chrominance de type 4:2:2, un pixel U et V pour deux pixels Y dans la direction horizontale et un pixel U et V pour chaque pixel Y en vertical. La plupart des modes de codage MPEG sont de type 4:2:0 (*un pixel de chrominance pour deux de luminance dans les deux directions*). Il est donc indispensable d'opérer par filtrage, puis sous-échantillonage, une conversion 4:2:2/4:2:0 des composantes chrominance.

La norme MPEG définit la position respective des échantillons de chrominance par rapport à la luminance par contre aucun filtrage n'est spécifié par le standard.

III.1.1.3.3 - Sous-échantillonnage (Subsampling) et filtrage

Le sous-échantillonnage est une technique simple qui revient à diminuer la taille (*verticale et/ou horizontale*) des images et par conséquent à diminuer le nombre de pixels à transférer.
L'image sera donc interpolée par le décodeur pour retrouver sa taille originelle.

Le filtrage permet de limiter les effets de blocs provoqués par une quantification (*voir plus loin*) trop élevée. On substitue donc des défauts de codage par un flou de l'image qui, en général, est mieux accepté par l'œil humain.

III.1.1.3.4 - Formatage des données

Avant compression, le signal est organisé par niveaux hiérarchiques.

Dans le cas d'un signal entrelacé, l'image peut d'abord être séparée en deux trames codées séparément.

L'**image** ou **trame** est constituée de **rangées horizontales**, chacune contenant 16 lignes de pixels.
Dans chaque rangée (ligne de pixels) on trouvera des macroblocs, chacun étant de dimension 16 x 16 pixels.

Un **macrobloc** est organisé en 4 blocs de luminances et en 2, 4 ou 8 blocs de chrominance selon le type d'échantillonnage. (4:2:0,4:2:2,4:4:4)

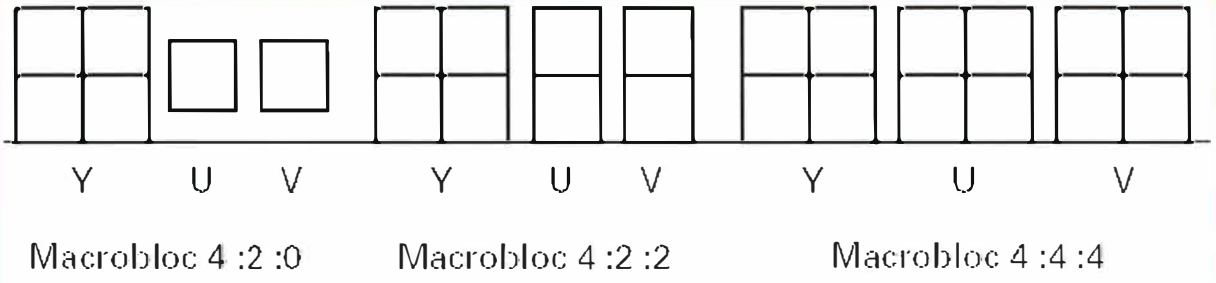


Figure III-1 : Différentes compositions d'un macrobloc

Dans un schéma de compression d'un signal entrelacé, on peut être amené à formater les données en mode image ou trame dans un macrobloc. Dans le premier cas, les deux parties du macrobloc correspondant aux deux trames restent entrelacées dans des blocs communs. Dans le mode de codage de type trame, les deux trames sont séparées dans le macroblocs : deux blocs de luminance correspondent à la trame paire et deux autres à la trame impaire.

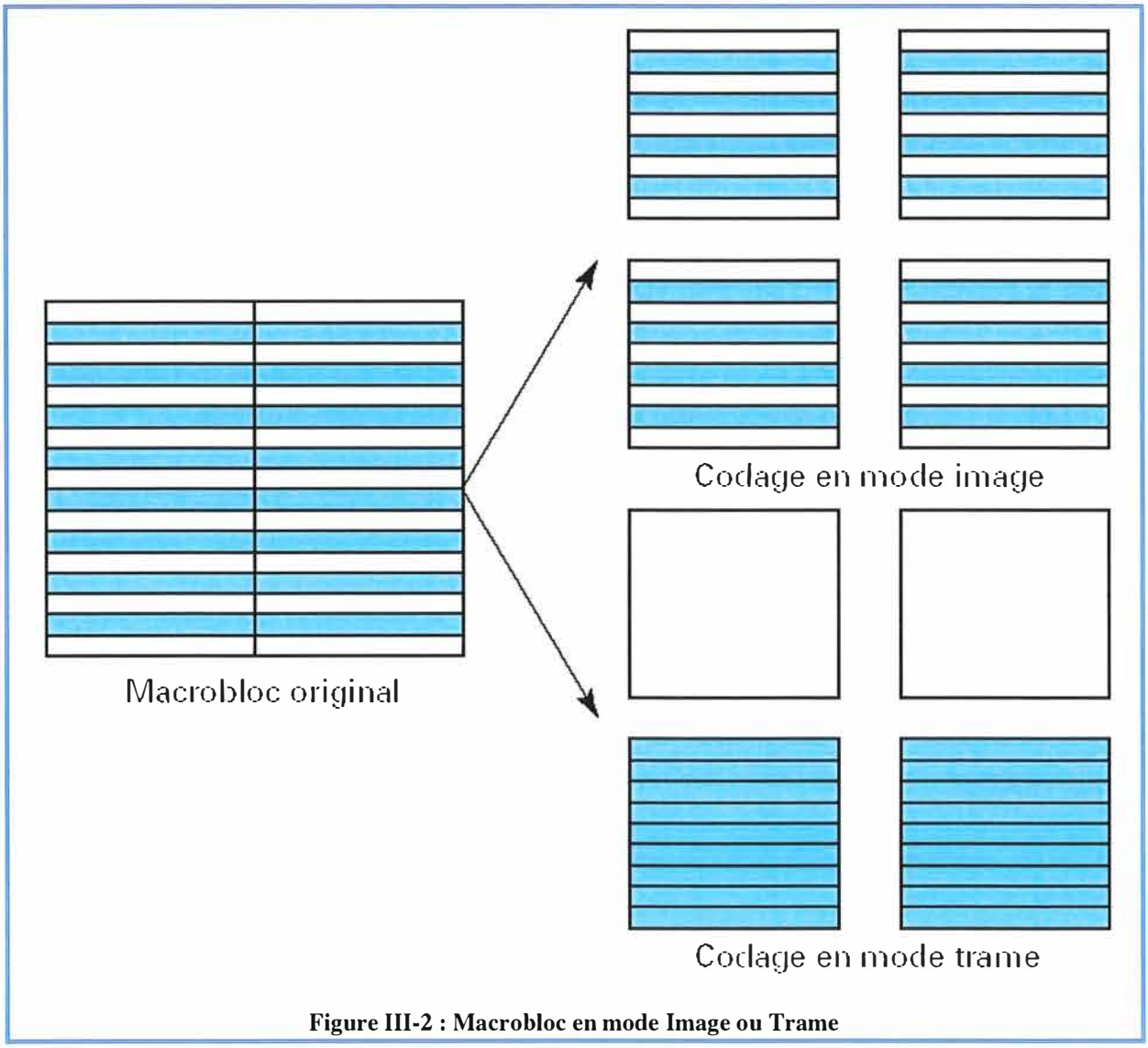


Figure III-2 : Macrobloc en mode Image ou Trame

Ce choix, quand il est autorisé, peut se faire à chaque macrobloc, un bit dans le flux binaire indiquant le mode sélectionné. En règle générale, il est préférable de passer en mode trame

pour une image comportant du mouvement, la cohérence entre les trames étant faible, alors que le mode image est plus efficace quand le mouvement est faible, car il existe une forte cohérence entre les deux trames.

III.1.1.3.5 - DCT et DCT Inverse

Comme vu précédemment, l'un des objectifs fondamentaux de la compression du signal vidéo est de réduire les redondances spatiales. La transformée en cosinus discret (***Discrete Cosines Transform, DCT***) est l'élément essentiel de ce processus.

Pour mieux comprendre voyons ce qu'est une *DCT* à une dimension

- *DCT à une dimension*

Une *DCT* à une dimension convertit un tableau de nombres (représentant l'amplitude d'un signal à différents instants dans le temps) dans un autre tableau de nombres représentant l'amplitude d'une certaine composante fréquentielle du tableau source.
On passe donc du domaine spatial au domaine fréquentiel.

- Le premier élément du tableau résultat, appelé coefficient *DC*, est une simple moyenne de tous les échantillons du tableau source.
- Les autres éléments, appelés coefficients *AC* indiquent chacun l'amplitude d'un composant fréquentiel spécifique du tableau source.

- *DCT à deux dimensions*

Cette fois-ci nous avons un tableau à deux dimensions comme source.
Dans le cas précis du MPEG, il s'agit d'un tableau de 8 x 8 pixels.
L'on va appliquer le *DCT* (de type une dimension) sur chacune des 8 lignes et donc obtenir 8 tableaux de coefficients.
En superposant ces 8 tableaux, on reforme un tableau de 8 x 8 pixels dont la première colonne contient tous les coefficients *DC*, la 2ème colonne contient le premier coefficient *AC* de chaque colonne, etc …
Le problème est que malgré le fait que le tableau en horizontal représente bien des informations fréquentielles, en revanche en vertical, il s'agit d'informations spatiales.
Donc il revient d'appliquer le *DCT* mais cette fois sur chacune des colonnes.

La DCT est une transformée orthogonale en fréquence définie comme suit :

$$f(x,y)=\frac{2}{N}\sum_{u=0}^{N-1}\sum_{V=0}^{N-1}C(u)\,C(v)\,F(u,v)\quad\cos\frac{(2x+1)u\pi}{2N}\quad\cos\frac{(2y+1)v\pi}{2N}$$

La transformée inverse est définie comme suit :

$$f(u,v)=\frac{2}{N}C(u)\,C(v)\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}f(x,y)\quad\cos\frac{(2x+1)u\pi}{2N}\quad\cos\frac{(2y+1)v\pi}{2N}$$

$$C(u),C(v)=\frac{1}{\sqrt{2}}\ \textit{pour}\ uv=0$$

$$C(u),C(v)=1\ \textit{pour u ou}\ v\neq 0$$

Un bloc de dimensions N x N est donc transformé dans la phase de codage en un bloc de mêmes dimensions contenant les coefficients DCT. Le coefficient de la première colonne et première ligne correspond à la composante continue appelée DC. Ensuite, chaque coefficient appelé AC représente la contribution du bloc dans la composante DCT correspondante. Le déplacement vers la droite indique une augmentation de la fréquence horizontale, de même pour la dimension verticale de haut en bas.

La transformée permet de concentrer l'énergie du bloc codé sur certains coefficients. Ses avantages comparés à d'autres transformées du même type (Hadamard, Fourrier …) résident dans les points suivants : *simplicité d'implémentation dans des systèmes numériques, bonnes performances en terme de concentration de l'énergie et résultats de la transformée en valeurs réelles*. Le choix de la dimension pour les codages de type MPEG (N = 8 x N = 8) provient d'un compromis entre les différents avantages et désavantages correspondant à des dimensions plus ou moins petites. L'augmentation de la taille du bloc tend à améliorer l'effet concentrateur d'énergie de la DCT. En revanche, dans le même temps, on observe des effets provenant des défauts de quantification plus gênants et la cohérence de l'information dans un bloc plus grand tend à diminuer.

Il est important de noter que la DCT en tant que telle n'est pas une opération de compression. Au contraire : les pixels en entrée sont codés sur 8 bits alors que la sortie est codée sur 11 bits pour le DC et 12 bits pour les coefficients AC. L'opération de compression se fait effectivement dans les étapes suivantes.

III.1.1.3.6 - Quantification

L'opération de quantification est la première étape du processus de compression de l'information. Le coefficient *DCT* est quantifié de façon à réduire la dynamique du signal à coder. Celui-ci sera restitué dans le décodeur avec une erreur de quantification qui est à l'origine des défauts visuels introduits par les schémas de compression de type MPEG. Le choix du pas de quantification est donc directement à l'origine du compromis qui doit être trouvé dans un codeur entre la qualité de restitution du signal et le débit numérique utilisé. Une quantification plus forte implique des défauts plus visibles mais un débit plus faible, et inversement.

Le processus de quantification n'est pas spécifié dans la norme. De plus ce processus de quantification ainsi que celui de quantification inverse dans le décodeur sont différents en fonction du type de coefficient *DCT*.

- *Coefficient DC*

Soit *quant (DC)*, la valeur quantifiée du coefficient *DC*, *quant $^{-1}$ (DC)* la valeur déquantifiée et *DC_quant_step* le pas de quantification, le processus de quantification/déquantification est le suivant :

$$quant\,(DC) = \frac{DC}{DC\_quant\_step}$$

$$quant^{-1}(DC) = quant\,(DC)\,x\,DC\_quant\_step$$

Le choix de *DC_quant_step* se fait pour chaque image et détermine directement le nombre de bits sur lequel est codé le résultat :

8 (*DC*_quant_step = 8) à 11 bits (*DC*_quant_step = 1).

- *Coefficient AC*

Soit *quant[AC(u,v)]* la valeur quantifiée du coefficient *AC(u,v)* de la colonne *u* et ligne *v* du bloc *DCT* et *quant* $^{-1}$ *[AC(u,v)]* la valeur obtenue après déquantification, le calcul de quantification se fait en deux étapes.

$$quant\,[AC(u,v)] = (16 * AC(u,v)) // weight(u,v)$$

Et ensuite pour les *blocs intra* :

$$quant\,[AC(u,v)] = \frac{\left\{ quant[AC(u,v))] + sign[AC(u,v))] * \left[\frac{(3 * mquant)}{4} \right] \right\}}{(2 * mquant)}$$

$$quant^{-1}[AC(u,v)] = \frac{(2 * quant[AC(u,v))]) * weight(u,v) * mquant}{32}$$

Et pour les *bloc Inter* :

$$quant\,[AC(u,v)] = \frac{quant[AC(u,v)]}{(2 * mquant)}$$

$$quant^{-1}[AC(u,v)] = \frac{((2 * mquant[AC(u,v)] + sign(quant[AC(u,v)]) * weight(u,v)) * mquant)}{32}$$

On remarquera dans cette formule de quantification que plusieurs paramètres influent sur le niveau de quantification.

- *mquant* : paramètre de quantification commun à tous les coefficients *DCT* d'un macrobloc. Il permet donc de faire varier le niveau de quantification à chaque macrobloc.

- *weight(u,v)* : dans chaque séquence de codage MPEG on fixe des matrices de quantification contenant ces poids correspondant à chaque fonction *DCT*. Cette différentiation permet de quantifier plus fortement les hautes fréquences. En effet, le système visuel humain est plus sensible aux défauts de codage dans les basses fréquences. De plus, deux matrices différentes peuvent être utilisées selon le type de codage utilisé : inter ou intra. Cette différenciation se justifie par le fait que le type de signal à coder après le processus de compensation de mouvement a des propriétés différentes d'un signal codé en intra.

- *Zone morte* (3 x mquant // 4 en intra) : cette valeur correspond à ce qui est communément appelé la zone morte. Ce décalage dans la courbe de quantification permet d'obtenir un nombre plus important de coefficients nuls après quantification, et donc de diminuer la quantité d'informations à coder.

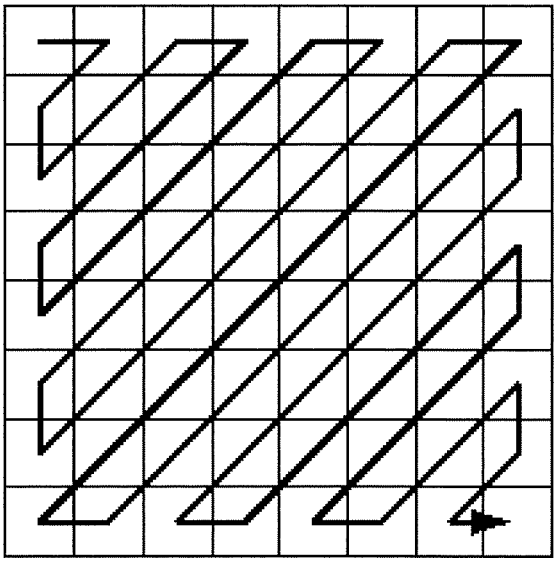
III.1.1.3.7 - Codage Statistique (Variable Length Coding)

Le processus de quantification a permis de réduire la quantité d'informations à coder. Néanmoins, c'est l'opération de codage statistique qui va mettre à profit les opérations de *DCT* et de quantification pour diminuer de façon significative la taille du flux binaire. L'opération de codage statistique permet de compresser le flux binaire décrivant les coefficients *DCT* en utilisant les propriétés statistiques du signal. On utilise un codage à longueur variable de type Huffman (*VLC - **V**ariable **L**ength **C**oding*) qui permet de coder les valeurs les plus probables avec les mots binaires les plus courts et les valeurs les moins probables avec les mots les plus longs. Les tables de correspondance entre valeurs quantifiées et mots *DCT* sont définies pour exploiter au mieux les statistiques d'un signal après quantification dans un codeur MPEG.

L'opération inverse de décodage à longueur variable (*VLD – **V**ariable **L**ength **D**ecoding*) dans les décodeurs permet de restituer les valeurs des coefficients *DCT*. De même pour l'opération de quantification, on distingue le codage des coefficients *DC* et *AC*.

DC : chaque coefficient *DC* est codé en mode différentiel (codage de la différence) par rapport aux coefficients DC précédant dans l'ordre de transmission. La valeur de la différence est codée en deux mots : le premier représente la taille (maximum de 8 à 12 bits selon la précision de codage choisie) et le second donne la valeur codée sur le nombre de bits correspondant à la taille.

AC : le tableau bidimensionnel issu du processus de quantification est d'abord transformé en tableau monodimensionnel en utilisant la technique du *ZigZag*.



Cette séquence a la propriété de parcourir les éléments en commençant par les basses fréquences et de traiter les fréquences de plus en plus hautes. Puisque la matrice *DCT* quantifiée contient beaucoup de composantes de hautes fréquences nulles, l'ordre de la séquence zigzag va engendrer de longues suites de 0 consécutifs.
Ensuite, on utilise un mode de codage dit *Run/length* qui permet d'exploiter au mieux la forte probabilité de présence de valeurs nulles dans le tableau. Pour chaque élément non nul du tableau, on code le nombre de zéros qui le précède ainsi que la valeur.
Quand on ne rencontre plus de valeurs non nulles, le mot de code indiquant la fin du bloc est envoyé (*EOB*) d'où l'intérêt de concentrer les valeurs non nulles en début de bloc.

Chaque couple est ensuite codé dans le flux par son mot correspondant dans la table *VLC*.

Si le couple ne fait pas partie de la table, un mot de code spécifique est envoyé (*escape code*), suivi de la longueur sur 6 bits et la valeur sur 12 bits.

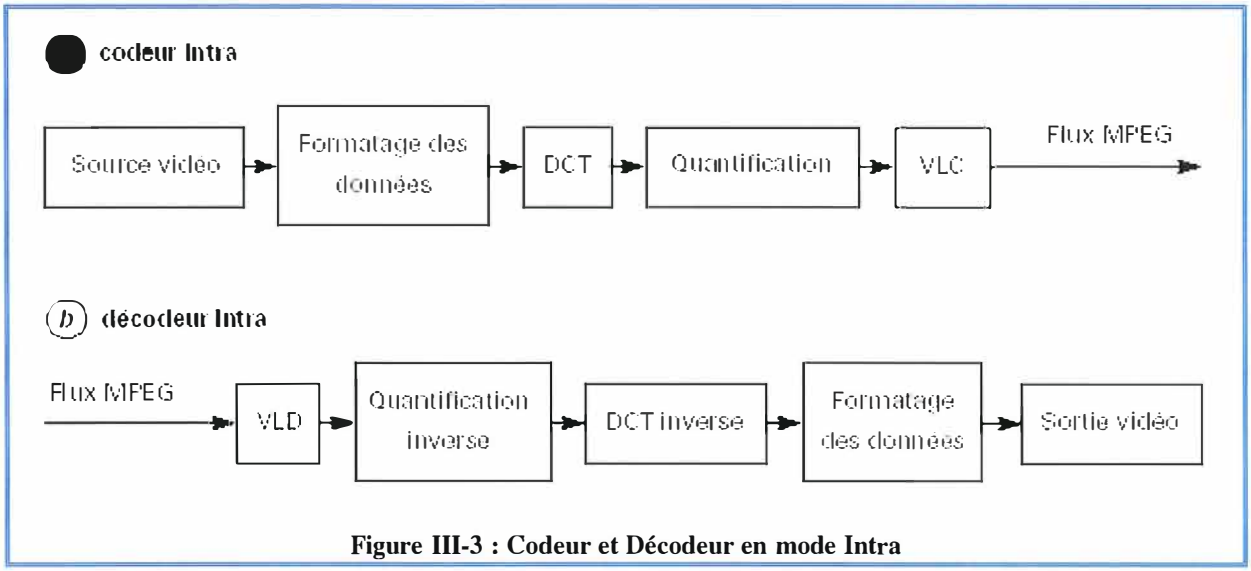


Figure III-3 : Codeur et Décodeur en mode Intra

III.1.1.4 - Type de Codage 'Inter'

III.1.1.4.1 - Estimation de mouvement et compensation

Le mode de codage Inter a pour but de mettre à profit les redondances temporelles du signal vidéo pour le compresser. Le principe est donc de prédire le contenu d'une image, puis de coder uniquement l'erreur faite sur cette prédiction. La méthode la plus simple est de faire la différence entre les valeurs de chaque pixel à position égale et ensuite de coder l'image différence. Cette opération est peu efficace si le contenu de l'image est en mouvement. Les normes MPEG mettent donc en œuvre des techniques de compensation du mouvement dans l'image pour optimiser la réduction des redondances temporelles. Plusieurs étapes sont alors à distinguer.

- *L'estimation de mouvement* : en règle générale, le mouvement dans une séquence vidéo ne peut pas se modéliser par un seul vecteur (sauf dans le cas d'un panning simple). A chaque *macrobloc* de l'image, on associe donc une information de mouvement. Dans les normes MPEG, seuls les mouvements de type translation sont modélisés : l'utilisation de mouvements de type homothétie ou rotation n'améliore pas suffisamment les performances de compression en regard de la complexité qu'ils induisent dans les systèmes de compression et décompression. L'opération d'estimation de mouvement permet de déterminer dans l'image de référence le macrobloc qui ressemble le plus au macrobloc à coder. Cet algorithme de recherche n'est pas normalisé et son efficacité a une influence fondamentale sur la performance du codeur, mais aussi sur sa complexité. La méthode la plus utilisée est le *blockmatching* : le macrobloc est comparé avec les macroblocs pointés par les vecteurs testés dans la zone de recherche de l'image de référence. Le vecteur est en général déterminé avec une précision d'un demi pixel. La sélection est faite sur le macrobloc minimisant la différence du point de vue de la somme des valeurs absolues des différences entre les valeurs de pixels.

- *La compensation de mouvement* : l'information de mouvement ayant été déterminée pour chaque macrobloc, on détermine le macrobloc qui représente la référence. Dans le codeur, l'extraction de ce macrobloc doit se faire dans l'image de référence décodée et non l'image originale, de façon à permettre la même opération de compression dans le décodeur où seules les images décodées sont disponibles. Dans le cas contraire, une dérive des images survient dans le décodeur au fil du décodage, seul le mode de codage Intra sans utilisation de la référence permettant de revenir à une référence commune.

- *Le codage* : le macrobloc de prédiction étant déterminé, il suffit d'en faire la différence avec le macrobloc à coder. De façon à permettre l'opération inverse dans le décodeur, l'information sur le vecteur mouvement utilisé sera codée dans le bitstream pour chaque macrobloc. Le macrobloc différence sera traité de la même façon qu'en mode Intra, avec quelques adaptations dues aux caractéristiques statistiques du signal.

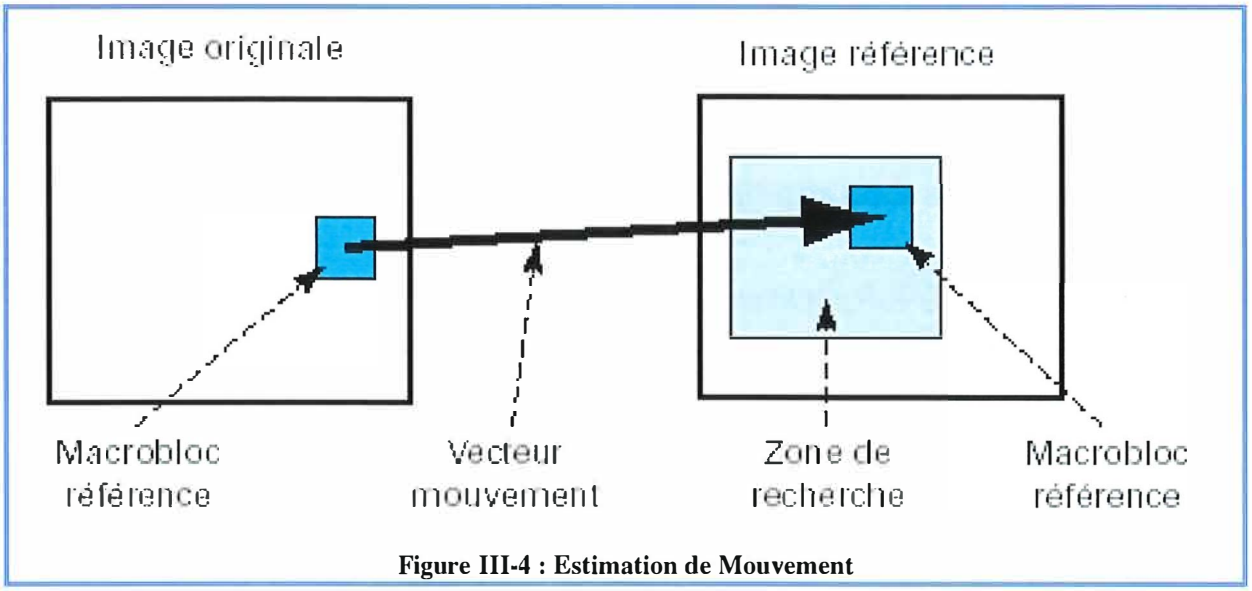


Figure III-4 : Estimation de Mouvement

III.1.1.4.2 - Images I, P, B

On distingue dans un flux MPEG plusieurs types d'images selon les modes de prédiction utilisés :

- *Image I* (***Intra Picture***) : dans cette image, les macroblocs sont codés en Intra, donc sans faire référence à une autre image. Ces images sont donc les points d'accès dans un flux MPEG pour le décodage. On notera que l'efficacité de la compression étant limitée à la réduction des redondances spatiales, les images I, à qualité égale, ont le taux de compression le plus faible (mais la taille la plus grande).

- *Image P* (***Predictive Picture***) : les macroblocs sont codés en mode Inter par rapport à une image P ou I précédente dans le flux vidéo.

- *Image B* (***Bi-directionally Predictive Picture***) : les macroblocs sont prédits par rapport à l'image P ou I précédente et l'image P ou I suivante. Cette possibilité est la plus efficace du point de vue de la réduction des redondances temporelles (une information non présente dans l'image précédente peut se trouver dans l'image suivante), et donc

ce type d'image contient la quantité la plus faible d'informations à qualité d'image égale. En revanche, la prédiction par rapport à une image future qui doit être préalablement codée implique un processus de réordonnancement des images aussi bien dans le codeur que dans le décodeur.

Deux paramètres caractérisent la structure d'un point de vue images I, P, B d'un flux MPEG.

- *N* représente la distance entre deux images I successives. L'augmentation de N implique une meilleure qualité de codage, en revanche, l'accès dans la séquence est plus restrictif (cet aspect est important dans les applications de télévision numérique où le zapping est une fonctionnalité importante).

- *M* représente la distance entre deux images P successives. L'augmentation de M permet une meilleure qualité de codage, mais s'accompagne d'un retard de codage/décodage et d'une complexité de réalisation plus importante.

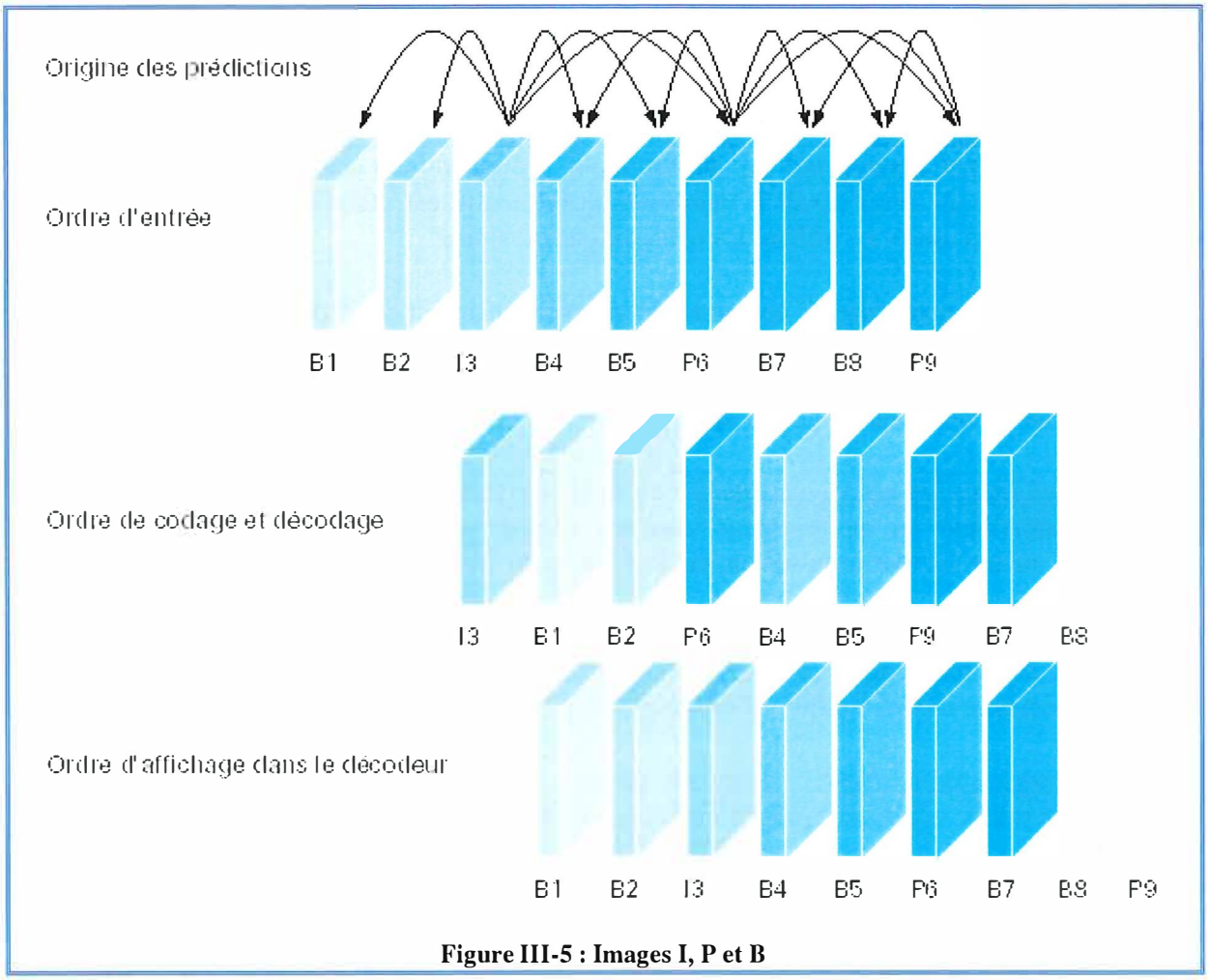


Figure III-5 : Images I, P et B

III.1.1.4.3 - Structure image ou trame

Afin de prendre en compte le caractère entrelacé du signal d'entrée, on définit les modes de codage en structure image ou trame.

- *Structure image* : les deux trames de l'image d'entrée sont traitées dans un seul élément syntaxique commun. Bien entendu, la structure entrelacée du signal peut toujours être prise en compte dans le codage d'un macrobloc.

- *Structure trame* : les deux trames sont traitées dans deux éléments syntaxiques différents, l'une après l'autre. En particulier, le type d'image peut être différent : une image est séparée en une trame impaire I et une trame paire P. C'est ici qu'apparaît le principal avantage de la structure trame : seule une trame est codée en mode Intra au lieu d'une image entière, ce qui permet d'améliorer l'efficacité de compression. Mis à part ce point très spécifique, le mode trame est peu utilisé, la cohérence entre deux trames d'une même image impliquant en général une meilleure efficacité du mode image.

III.1.1.4.4 - Modes de compensation

Après avoir choisi un type d'image, la norme de type MPEG laisse aussi un choix étendu sur le mode de compensation pour chaque macrobloc, l'entrelacement de l'image d'entrée étant pris en compte.

- *Compensation image* : c'est le mode de compensation le plus naturel. Le vecteur mouvement correspond au déplacement du macrobloc dans une image avec une précision au demi pixel. L'interpolation bilinéaire est utilisée pour évaluer la valeur des pixels situés au milieu des échantillons du signal.

- *Compensation trame* : ce mode de compensation est utile pour les sources de type entrelacé en mouvement. Un vecteur mouvement est utilisé pour chaque trame du macrobloc. La réduction se fait dans l'une des deux trames de l'image de référence toujours avec une précision demi pixel.

- *Compensation sans mouvement* : ce mode existant uniquement pour les images P est l'équivalent du mode image avec un vecteur nul. Son intérêt réside dans le fait qu'aucun vecteur mouvement ne doit être transmis dans le flux.

- *Compensation dual prime* : ce mode met en œuvre une compensation tenant compte du caractère entrelacé du signal en transmettant un seul vecteur. Il est autorisé uniquement dans un flux ne comportant pas d'images B.

- *Compensation 16x8* : dans les images codées en structure trame, la structure macrobloc correspond spatialement à une taille dans l'image de 16x32 . Le mode 16x8 permet d'utiliser deux vecteurs mouvements pour se ramener à une taille de macrobloc en mouvement plus naturelle.

- *Modes interpolés* : les images B autorisent les modes de compensation par rapport aux images P précédentes et suivantes. Dans le mode interpolé, une pondération est faite entre ces deux prédictions, et donc, par effet de filtrage temporel, on obtient un macrobloc de référence plus proche de l'original.

- *Compensation 8x8* : autorisé dans MPEG-4 uniquement, ce mode permet une compensation plus fine, grâce à l'utilisation d'un vecteur mouvement par bloc.

- *Compensation du mouvement global* (*GMC - **G**lobal **M**otion **C**ompensation*) : prévu par MPEG-4 uniquement, ce mode compense chaque bloc en utilisant des paramètres globaux, valables pour tous les blocs d'un même objet. La nature de ces paramètres et la technique de compensation à leur associer sont les mêmes que celles utilisées pour les sprites.

- *Mode Intra* : si aucun des modes précédents n'est satisfaisant du point de vue de la ressemblance, il est toujours possible de coder le macrobloc en mode Intra sans faire référence à une autre image.

III.1.1.4.5 - Sélection du mode

La norme permet de sélectionner pour chaque macrobloc l'un des modes de compensation décrits dans le paragraphe précédent, mais ne précise pas le critère du choix. En général, on reprend le même type de critère que celui utilisé dans le processus d'estimation de mouvement

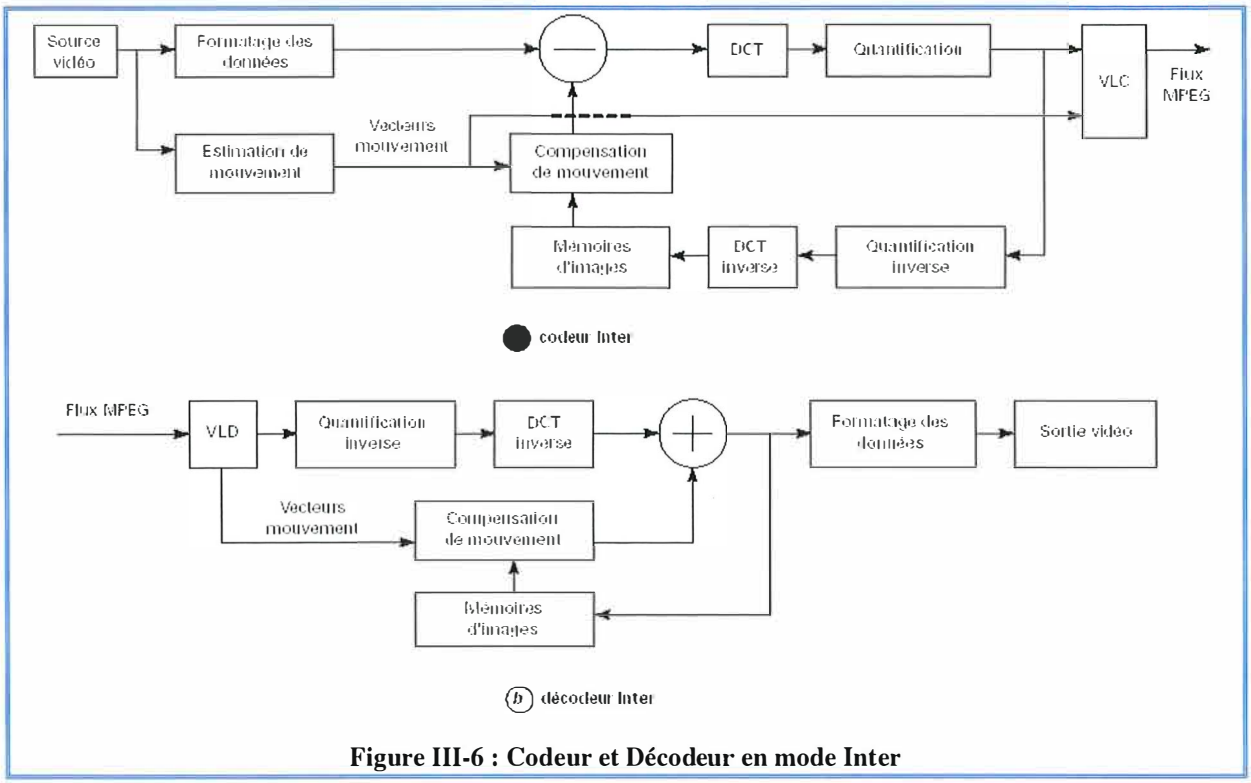


Figure III-6 : Codeur et Décodeur en mode Inter

III.1.1.5 - Organisation d'un flux MPEG

Les normes de compression MPEG définissent l'organisation d'un flux binaire ainsi que la signification de chaque élément. Le séquencement exact, bien que suivant certaines règles de base, n'est pas complètement défini et dépend des choix de codage effectués par le codeur (N, M, modes de compensation…). Chaque élément syntaxique de haut niveau (*séquence, image, rangée*) est précédé d'un code d'accès unique : 23 zéros suivis d'un 1. Ceci permet à un décodeur de retrouver rapidement une synchronisation en début de décodage ou quand une erreur est survenue sur le flux binaire.

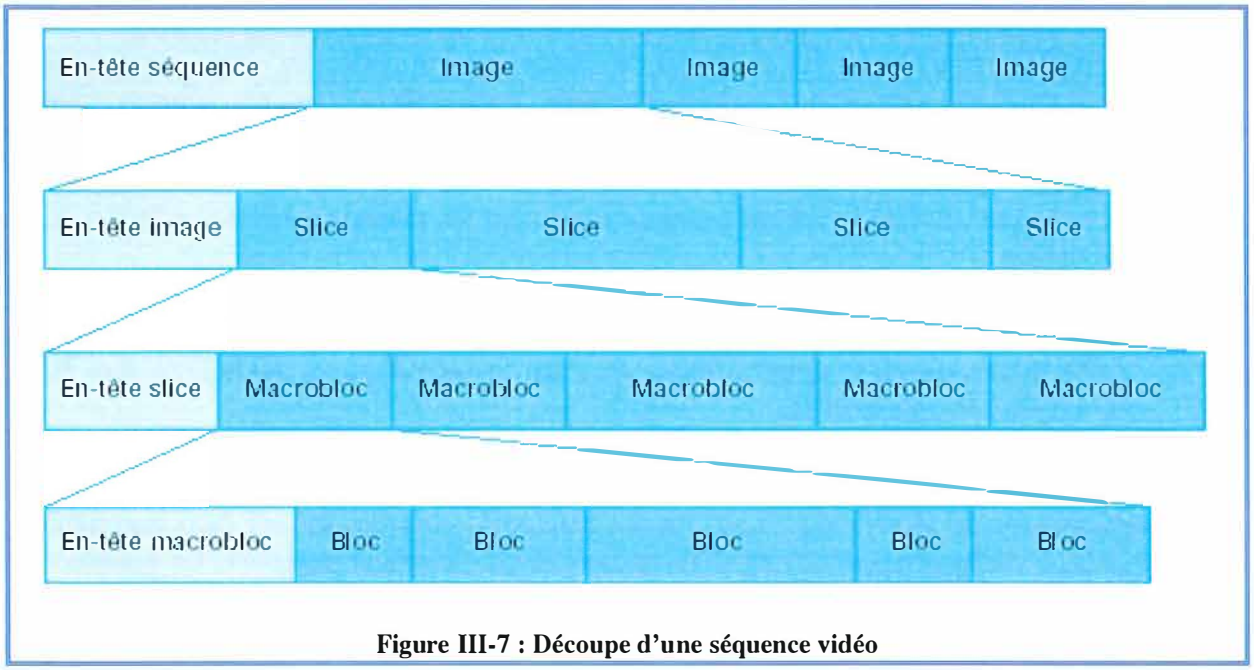


Figure III-7 : Découpe d'une séquence vidéo

ou encore

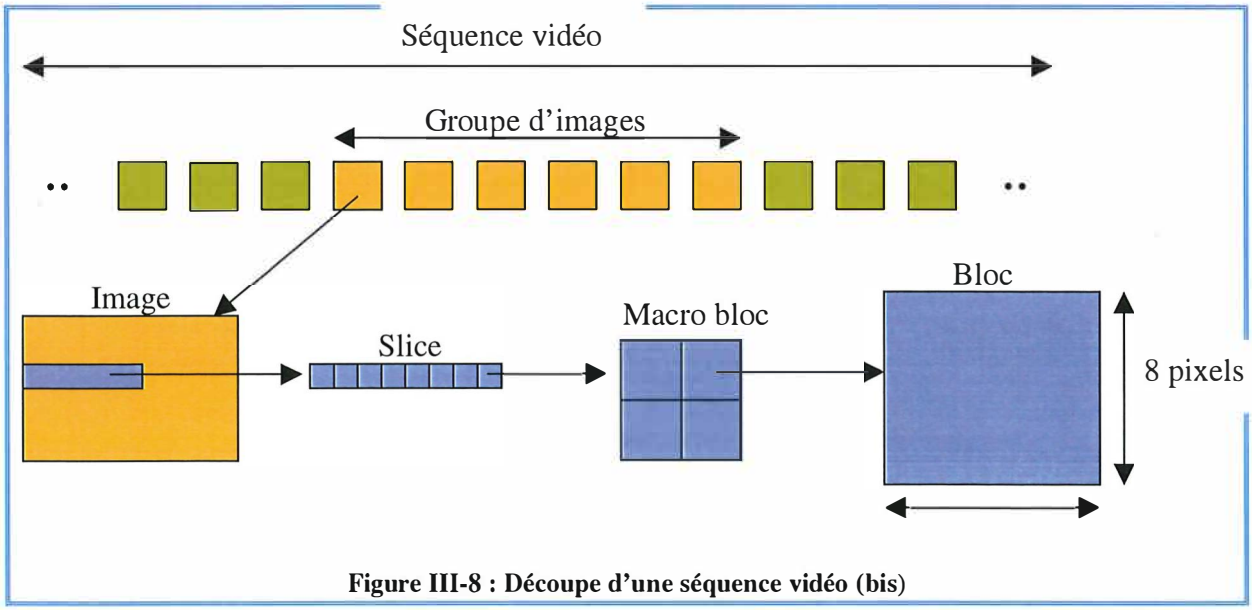


Figure III-8 : Découpe d'une séquence vidéo (bis)

III.1.1.5.1 - Séquence Vidéo

Une séquence vidéo représente un certain nombre d'images vidéo ou groupe d'images vidéo. Même si le nom peut donner une autre impression, une séquence vidéo ne contient seulement que quelques images et pas un film entier.

Chaque flux binaire MPEG doit débuter par un en-tête de séquence. Les paramètres de codage, invariables au cours d'une séquence, y sont indiqués : *fréquence image, taille de l'image, débit binaire, matrices de quantification.* Cet en-tête devra obligatoirement être suivi d'une image I pour débuter le décodage. Dans les applications de transmission de télévision

numérique, cet en-tête sera répété régulièrement de façon à garantir à l'utilisateur un accès aléatoire dans le flux reçu. Le flux se terminant par un code de fin de séquence

III.1.1.5.2 - Groupe d'Images (Group of Pictures – GOP)

Il regroupe un en-tête et une série d'une ou plusieurs images permettant d'y accéder de façon aléatoire.

III.1.1.5.3 - Image (Frame ou Picture)

Une image contient toutes les informations de couleur et de luminance nécessaires pour jouer une image à l'écran. Les informations de couleur et de luminance sont réparties en 3 matrices, qui contiennent des valeurs de luminance et de chrominance. La taille de ces matrices varie selon la résolution d'image choisie et le ratio d'échantillonnage utilisé.

Chaque image transmise dans l'ordre de codage (qui n'est pas toujours l'ordre de réception dans le codeur) est précédée d'un en-tête contenant les informations générales spécifiques pour le décodage de l'image : *type d'image, structure de l'image, dimension maximale des vecteurs mouvement…*

III.1.1.5.4 - Slice

Une image est constituée d'un ensemble de rangées de *macroblocs*, marquées en leur début par un en-tête permettant la resynchronisation du décodeur en cours d'image. La fréquence d'insertion de ces en-têtes peut augmenter toujours pour faciliter la resynchronisation en cas d'erreur sur le flux binaire. Chaque élément syntaxique situé entre deux en-têtes successifs est appelé *slice* et contient un nombre variable de macroblocs.

Ce sont des éléments importants pour la gestion des erreurs. Si le flux de données contient une erreur, le décodeur peut sauter la tranche et passer au début de la suivante directement. Plus il y a de tranches, meilleur est le traitement des erreurs, mais cela fait perdre de la place.

III.1.1.5.5 - Macrobloc

C'est une matrice rectangulaire de dimension 2 et constituée de blocs.

Dans le flux correspondant à un macrobloc, on trouvera d'abord les informations nécessaires à son décodage : *pas de quantification, modes de codage, mode de compensation de mouvement et vecteurs mouvement*. Pour certaines de ces informations (*modes de compensation, vecteurs mouvement*), la norme fait appel à des codes à longueur variable. L'appel à des techniques de codage équivalentes aux codes de Huffman utilisés pour les coefficients *DCT* permet d'exploiter les statistiques d'occurrence des modes de compensation ou vecteurs. Par exemple, dans une image P ou B, le mode Intra est très peu probable.

III.1.1.5.6 - Bloc

Le bloc est l'unité élémentaire pour le codage de la séquence vidéo. C'est un ensemble des valeurs de luminance et chrominance de 8 lignes de 8 pixels.

Pour chaque macrobloc, on transmet les 6, 8 ou 12 blocs *DCT* correspondants dans l'ordre défini par la norme.

III.1.1.6 - Le MPEG-1 en concret (VIDEO)

La norme MPEG-1 utilise les fonctionnalités essentielles d'une compression vidéo à base de DCT et compensation de mouvement.
Le format généralement utilisé est le SIF (*quart d'image TV*) avec un débit de 1,25 Mbit/s. Puisque le SIF est un format progressif (une *image comporte les données prises par la caméra à un instant unique*), les caractéristiques spécifiques d'un signal entrelacé ne sont pas prises en compte ; cette limitation du standard implique simplement des performances de compression réduites si l'utilisateur transmet des images de type TV.

III.1.2 - MPEG-1 Partie 3: Audio (ISO/IEC 11172-3)

Les normes de compression audio MPEG définissent le processus de décodage d'un signal audio. De même que pour le signal vidéo, cette définition implique certaines lignes de conduite à respecter pour la compression.

III.1.2.1 - Modèles acoustiques

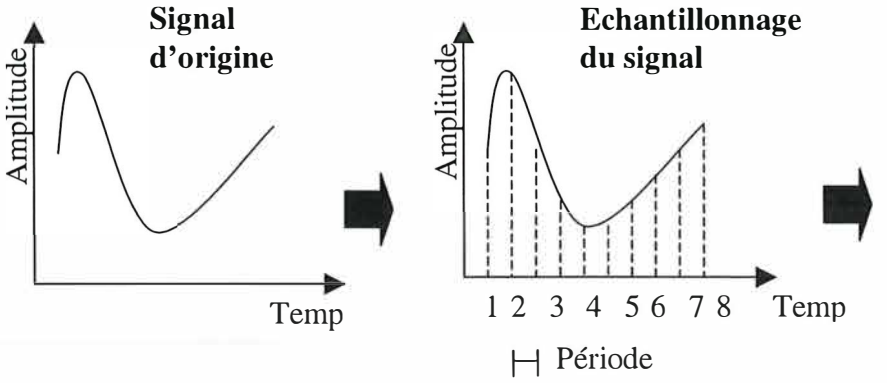
La base algorithmique de la compression audio MPEG est le système acoustique humain, qui n'a pas les mêmes caractéristiques qu'un instrument d'enregistrement. L'oreille humaine est un système *non linéaire à seuillage adaptatif*. En premier lieu, ce *seuillage* (non sensibilité à certains sons en deçà d'une puissance donnée) est variable en fonction de la fréquence, le maximum de notre sensibilité se situant en général entre 2 et 5 kHz. Ce modèle est compliqué par un *phénomène de masquage*. En effet, notre oreille percevra certains niveaux sonores assez bas dans un silence total, alors qu'un signal sonore comportant des fréquences similaires masquera l'audibilité des mêmes sons. Le mode de compression MPEG met donc à profit ces caractéristiques pour dédier la bande passante numérique aux sons audibles par une oreille humaine.

III.1.2.2 - Codage sous-bandes perceptuel

L'objectif de prise en compte des caractéristiques auditives de l'oreille est réalisé par l'utilisation d'un mode de codage par sous-bandes. Pour chaque sous-bande, le signal numérique d'entrée est traité par un filtre spécifique qui permet d'obtenir la composante sur cette bande de fréquence. Ensuite, chaque signal représentant la sous-bande est quantifié avec un pas dépendant du niveau de seuillage de la fréquence traitée. Le processus qui détermine le pas de quantification pour chaque sous-bande fait appel à un modèle psychoacoustique. Le choix de ce modèle détermine la qualité du codeur ainsi que sa complexité, les autres fonctions se retrouvant à l'identique dans chaque codeur audio. Cette opération permet de supprimer dans le signal les informations les moins perçues par l'oreille humaine. On transmet donc dans le flux MPEG les valeurs quantifiées ainsi que le pas de quantification utilisé dans chaque bande de fréquence. Le décodeur, après démultiplexage des données, quantification inverse et filtrage inverse, pourra reconstituer le signal décodé.

III.1.2.3 - Représentation digitale du son

Echantillonnage et conversion A/D d'un signal



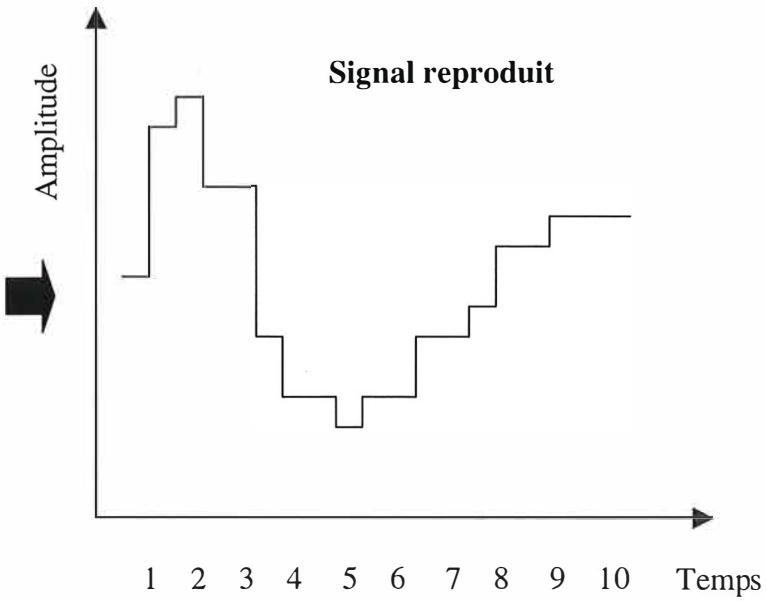
Valeurs binaires

No échantillon	Valeur
1	1000
2	1100
3	1110
4	1110
5	1100
6	0111
7	0011
8	0001
9	0001
10	0011

Conversion D/A d'un signal sous forme binaire

Valeurs binaires

No échantillon	Valeur
1	1000
2	1100
3	1110
4	1110
5	1100
6	0111
7	0011
8	0001
9	0001
10	0011



III.1.2.4 - Flux audio

Le flux audio MPEG est organisé en trames contenant un nombre fixe d'échantillons d'entrée (384 ou 1152). Aucune correspondance n'existe entre la durée des images vidéo et les trames audio. Au début de chaque trame, on trouve un en-tête avec un mot de signalisation et les informations de haut niveau nécessaires au décodage de la trame : *fréquence d'échantillonnage du signal d'entrée, débit de sortie compressé, mode de codage u*tilisé. On trouve ensuite les valeurs du signal d'entrée après filtrage et quantification.

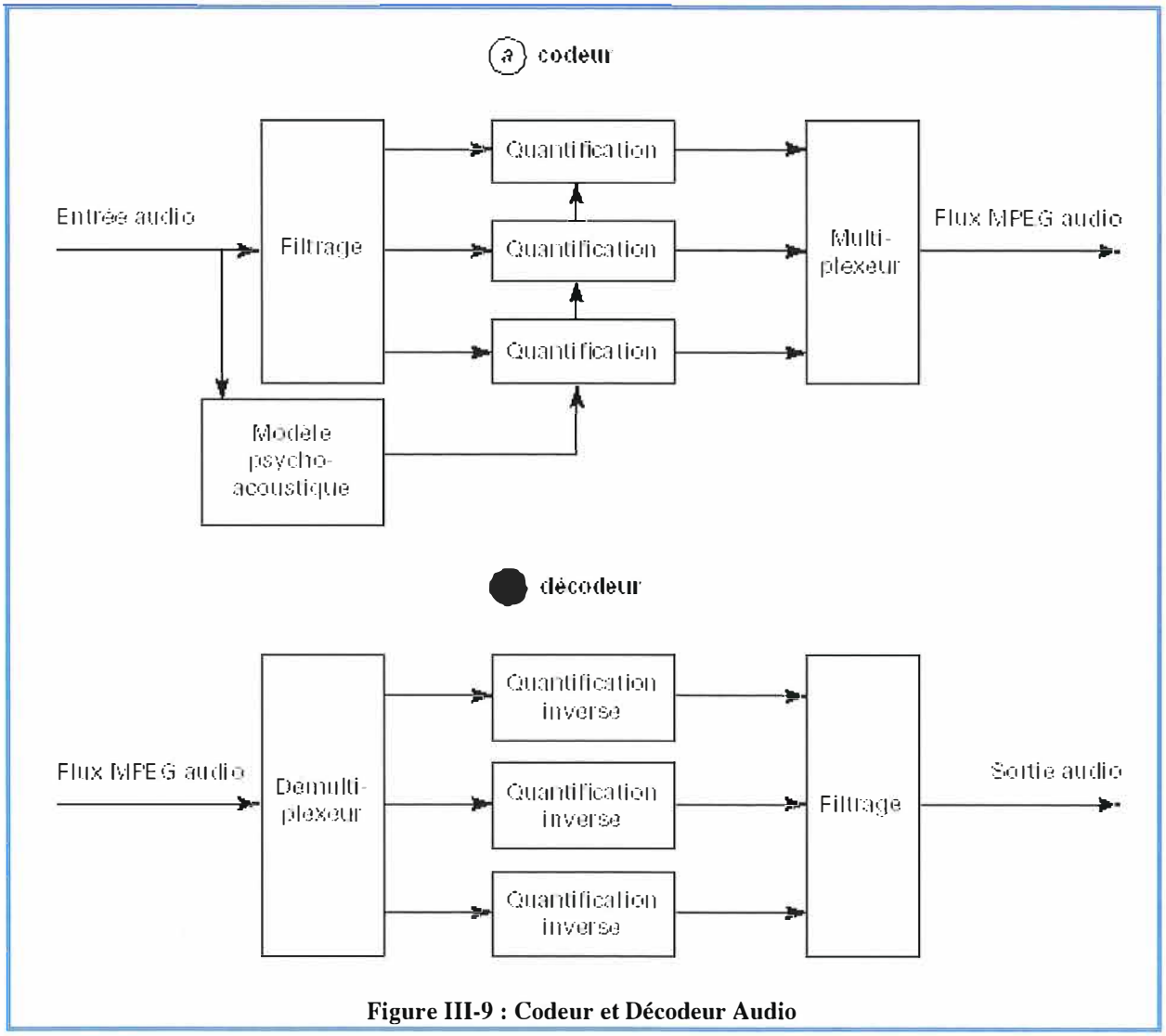


Figure III-9 : Codeur et Décodeur Audio

III.1.2.5 - Le MPEG-1 en concret (AUDIO)

Les fréquences d'échantillonnage autorisées vont de 32 à 48 kHz. Les débits varient entre 32 et 384 kbit/s. Trois niveaux (layers) de codage audio sont utilisés. Ces niveaux se distinguent par les outils de compression, les fréquences d'échantillonnage du signal d'entrée et les débits.

- *Niveau 1*

Le filtrage d'entrée est de type DCT avec utilisation d'un modèle psychoacoustique uniquement en fréquence.

- *Niveau 2*

Le filtrage d'entrée est aussi réalisé dans le domaine temporel, ce qui permet un certain masquage temporel.

- *Niveau 3*

Le filtrage d'entrée est modifié pour obtenir des largeurs de bandes de fréquences inégales et donc mieux adaptées au système auditif humain. Pour le cas d'un codage de signal stéréo, la cohérence entre les deux sources est utilisée. MPEG-1 audio niveau 3 est plus connu sous l'appellation MP3, qui est souvent déformée à tort en MPEG-3.

III.1.3 - MPEG-1 Partie 1: Système (ISO/IEC 11172-1)

III.1.3.1 - Multiplexage

Chaque flux élémentaire ayant été compressé séparément, les normes MPEG définissent des processus pour multiplexer ces données dans un seul flux à des fins de stockage ou de transmission. Encore une fois, seul le processus de décodage avec la signification de chaque bit est défini dans la norme.

La sortie d'un multiplexeur de type MPEG est un flux d'octets à un débit total fixe ou variable. Il existe plusieurs types de flux, l'application visée étant le critère essentiel de choix entre les options.

III.1.3.2 - Flux programme

Le flux de type programme est essentiellement spécifié pour répondre aux besoins d'une application de stockage. Dans cette optique, un seul programme (vidéo et audio) est multiplexé dans un flux. La spécification de ce flux répond essentiellement aux besoins suivants : synchronisation audio/vidéo, prévention des underflow ou overflow de la mémoire tampon, accès aléatoire aux données. Le formatage des données audio et vidéo dans des paquets *PES* (***P**acketized **E**lementary **S**tream*) permet d'assurer la synchronisation des données. En effet, on insère dans les en-têtes de paquets des estampilles temporelles qui spécifient les moments de traitement des données contenues dans le paquet. Le *DTS* (***D**ecoding **T**ime **S**tamp*) indique l'instant de décodage des données tandis que le *PTS* (***P**resentation **T**ime **S**tamp*) indique leur instant de présentation. Toutes ces indications sont données en unités d'horloge 90 kHz. Ces deux types de paquets *PES* sont ensuite multiplexés dans un même flux avec des en-têtes pour indiquer leur nature et les caractéristiques essentielles du signal. Le fonctionnement du décodeur de référence étant spécifié, ainsi que la taille de la mémoire

III.1.3.3 - Au niveau du Système

Seul le flux de type programme est spécifié dans la norme MPEG-1.

III.2 - LES APPLICATIONS DE MPEG-1

III.2.1 - Le VCD (Video CD)

(Informations venant du site http://www.media-video.com)

III.2.1.1 - Un peu d'histoire

Le VCD (*alias **VideoCD***) à été l'un des premiers supports numériques grand public à voir le jour. Objet de convoitise pour les premiers véritables amateurs de Home-Cinéma il se présentait sous la forme de gros disques lasers (*LD*) double face, simple couche. S'il ne mobilise plus les foules aujourd'hui en Europe il est encore très présent sur le marché asiatique (*Hong-Kong étant l'un des plus gros centres de vente*).

III.2.1.2 - Ces caractéristiques

Un VCD peut contenir environ 70 minutes de vidéo. Par sa faible résolution et son mode de compression (*le MPEG-1*) il montre vite ses limites. L'image manque de netteté, présente des artefacts (*blocs de pixels*) dans les scènes rapides et le son ne peut être encodés au mieux que en *MP2 Dolby Surround.*

Cependant, il présente l'immense avantage de pouvoir être lu dans presque tous les matériels équipés d'un lecteur CD récent. En effet, il peut être aussi bien lu sur un ordinateur, une console de jeux (*Playstation 1&2, Saturn & Dreamcast*), un lecteur DVD (*suivant les modèles*) ou encore sur les baladeurs CD/MP3 (*en le connectant à la télévision*).

Il est à noter que l'on peut également ajouter de l'interactivité avec, notamment, des menus (*au même titre que ceux des DVD*) et des sous-titres (*le "multi-langues" n'étant pas possible faute de place*).

Standard	Video-Cd (VCD)
Compression	MPEG-1
Bitrate (mbps)	Constant 1.15
Résolution en pixels **PAL** **NTSC**	 352 x 288 352 x 240
Images par seconde (frame per second) **PAL** **NTSC**	 25 fps 29.97 fps
Audio **MPEG-1, layer II (kbps)** **AC3 Dolby 5.1**	 224 non
Options Spéciales **Menus** **Multi-langues**	 Oui Non (mais sous-titres)
Espace disque	

1 seconde a/v	~180 kb
1h30min de film	~970 Mb
Utilisation	
PC DVD-Roms	Tous
Hi-FI DVD	Tous
PC CD-ROMs	Oui
SVCD Lecteurs	Oui

III.2.2 - Le CDI

Le CDI fut présenté par Philips comme une machine révolutionnaire, permettant de contenter tout le monde, les enfants pouvaient jouer, les adultes se cultiver au moyen de CDI documentaires et culturels ou se divertir en regardant des films.

Malheureusement, le CDI ne connu pas le succès escompté, et fut un véritable échec commercial. En effet très peux d'éditeurs ne furent intéressés par cet appareil aux caractéristiques techniques par trop limitées.

PARTIE 4

LE MPEG-2

IV - MPEG-2 (ISO/IEC-13818)

IV.1 - ANALYSE TECHNIQUE MPEG-2

MPEG-2 est une norme en 9 parties. Les parties audio, vidéo, système et DSM-CC seront analysées en détail dans la suite de ce document. La partie système prend en charge le multiplexage et la synchronisation de flux tandis que la partie DSM-CC s'attache à résoudre des problèmes plus orientés réseaux.

IV.1.1 - MPEG-2 Partie 2: Vidéo (ISO/IEC 13818-2)

IV.1.1.1 - Introduction

Cette partie est construite sur les puissantes capacités de compression de la norme MPEG-1. Le but principal de MPEG-2 Vidéo est de définir un format de description d'un flux de bits représentant de la vidéo codée. Le flux de bits vidéo est l'output d'un processus d'encodage, qui compresse les images vidéo de façon significative. MPEG-2 ne définit pas la méthode d'encodage, il définit uniquement le flux de bits résultant. Par contre, il définit comment décoder ce flux de bits. A première vue, cela peut sembler problématique que MPEG-2 ne spécifie pas le processus d'encodage. Néanmoins, cela permet au processus d'être ouvert à de futures améliorations, comme par exemple la réduction du temps d'encodage ou l'amélioration de la qualité de l'image.

Quand la norme MPEG-2 a été développée, une des exigences était de la rendre assez flexible pour servir à une vaste gamme d'applications. Les services (satellite) *broadcast*, la distribution TV par le câble et les services de télévision interactive sont autant d'exemples des applications envisagées.

Pour atteindre cette flexibilité, MPEG-2 doit pouvoir supporter différentes résolutions vidéo et différentes qualités d'images, et doit pouvoir s'adapter aux capacités des équipements et aux contraintes de bande passante induites par le réseau.

Le groupe MPEG-2 est parvenu à concevoir une norme très générique en fournissant un ensemble d'outils pouvant être combinés de multiples manières.

IV.1.1.2 - Syntaxe des flux de bits vidéo MPEG-2

Etant donné que MPEG-2 doit s'adapter à un grand nombre d'applications, il a fallu développer une syntaxe variable. Cela signifie que certains éléments de la syntaxe contrôlent l'apparence d'autres éléments de la syntaxe. En d'autres termes, certains éléments sont optionnels et sont présents uniquement si des flags (présents la plupart du temps dans l'en-tête) les signalent. Cela permet de réduire la somme des données à transmettre : au lieu de transmettre des valeurs vides comme des 0 ou des codes spéciaux, certains éléments ne sont pas présents dans le flux de bits.

IV.1.1.2.1 - Syntaxe Hiérarchique

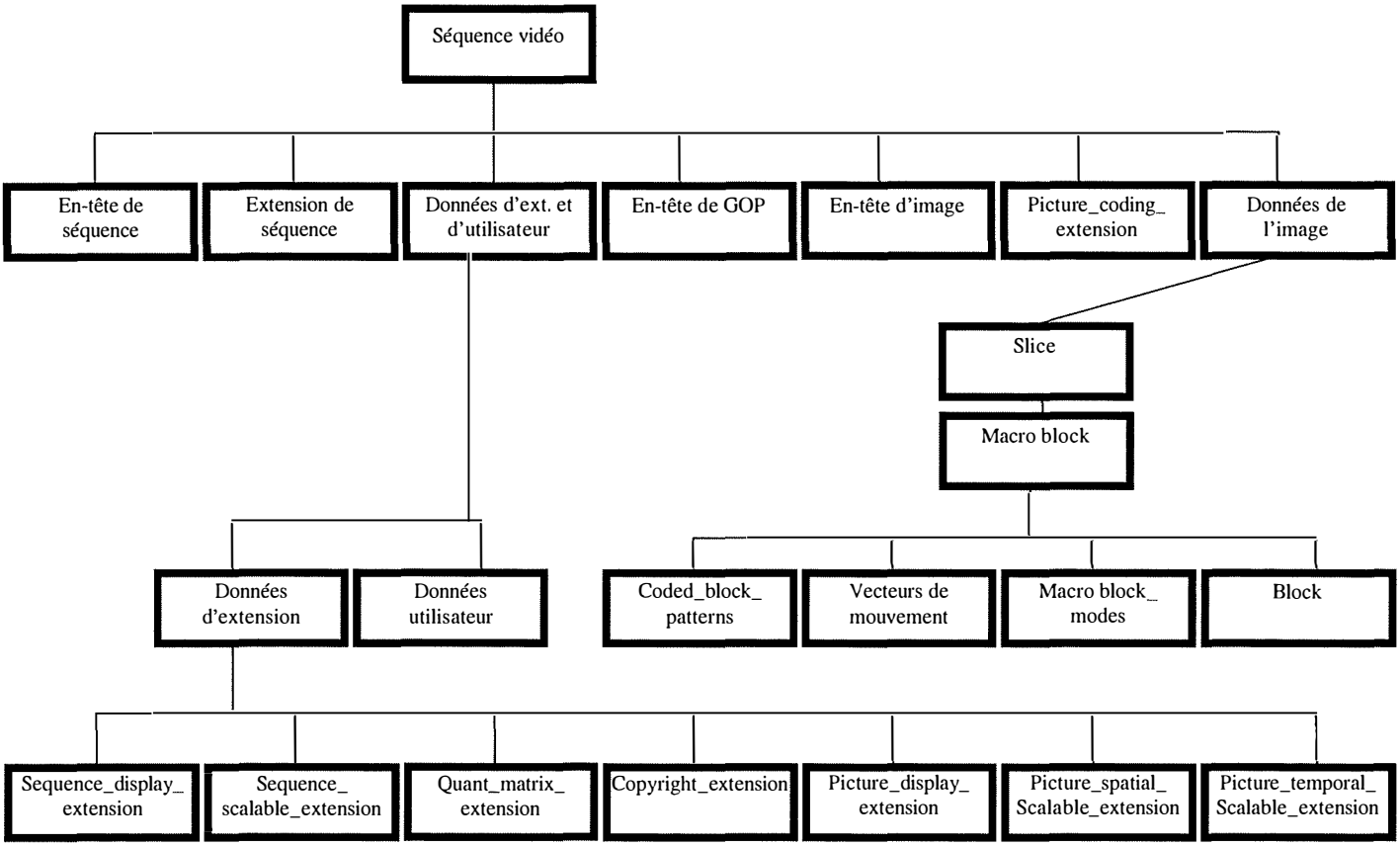


Figure IV-1 : Syntaxe vidéo hiérarchique

IV.1.1.2.2 - Détail de certains champs

L'*En-tête de séquence* contient les informations concernant la taille de l'image et le nombre d'images par seconde. L'*Extension de séquence* contient le profil, le niveau et le format de chrominance du flux de bits. Si ce champ n'est pas présent, le flux de bits est un flux MPEG-1 Vidéo. L'*En-tête d'image* indique notamment si l'image est de type I, P, ou B. Le *Mode macro bloc* indique la façon dont le macro bloc est encodé. Dans une image prédite, un macro bloc peut être encodé comme un macro bloc intra ou comme un macro bloc prédit. Dans une image bidirectionnelle, un macro bloc peut être encodé comme un macro bloc intra, prédit ou bidirectionnel. Ce champ indique également le mode de prédiction utilisé pour le macro block bidirectionnel. Un *Block* contient des coefficients DCT. Le *Copyright_Extension* indique si le flux est un original ou une copie et si le flux est protégé par des droits (et dans ce cas fournit le numéro de copyright).

Pour un détail de tous les champs, voir Annexe MPEG-2.

IV.1.1.3 - Scalabilité[1]

IV.1.1.3.1 - Notion

[1] *Note au lecteur: le terme scalabilité est la francisation pragmatique du terme anglais 'scalability'.*

Une des caractéristiques les plus importantes de la norme MPEG-2 Vidéo est sa capacité à supporter un large éventail d'applications vidéo différentes. MPEG-2 peut être utilisé pour la télédistribution standard, pour la télévision haute définition, ou pour la transmission vidéo via des réseaux de télécommunication. Au lieu de définir une spécification pour chaque application et d'avoir des formats de flux de bits différents pour chaque application, MPEG-2 utilise une approche dite de scalabilité. La scalabilité (la faculté d'adaptation) se concrétise à travers la syntaxe MPEG-2 Vidéo. L'information vidéo peut être divisée en différents flux d'information, qui sont complémentaires les uns par rapport aux autres. En fonction des applications, les flux d'information seront combinés de différentes manières. On utilise le terme de *couche* pour les différents flux d'information. Ci-dessous sont présentés les différents modes de scalabilité.

IV.1.1.3.2 - Scalabilité Spatiale

C'est la capacité de supporter plusieurs niveaux de résolution d'images dans un flux vidéo unique.

L'origine de ce mode est le déploiement des services de télévision numérique haute définition (HDTV) qui nécessite dans un premier temps d'offrir pour un même programme les sources en format TV et HDTV, ceci afin de permettre une migration progressive du parc des récepteurs. En pratique, on a une couche, appelée *couche de base*, qui contient l'information vidéo pour un programme TV standard (PAL ou NTSC), qui peut être combinée avec un autre flux d'information, la *couche additionnelle*, qui contient les informations vidéo supplémentaires pour obtenir une qualité vidéo de type HDTV. Selon les caractéristiques du décodeur, l'utilisateur sera capable de voir la télévision standard ou un programme en HDTV. Il est à noter qu'un seul flux de bits est délivré à l'utilisateur.

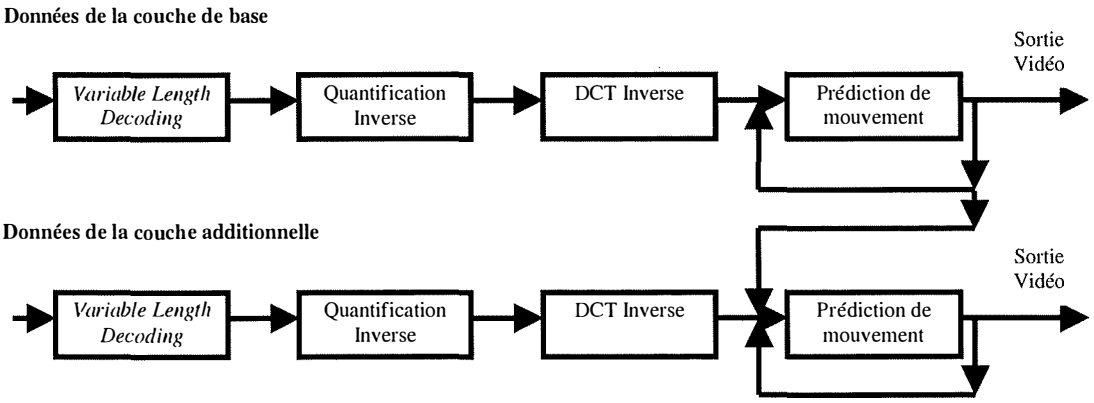


Figure IV-2 : Schéma de décodage avec la scalabilité spatiale

Avec la scalabilité spatiale, les données de la couche additionnelle et de la couche de base sont combinées après l'étape *DCT Inverse.* Le processus principalement affecté ici est la compensation de mouvement, qui peut utiliser les vecteurs de mouvement de la couche additionnelle ou de la couche de base.
Ce système est assez complexe à mettre en œuvre. Au regard du coût de cette fonctionnalité, le gain en compression reste assez faible par rapport à un système de transmission séparée des deux signaux. Ceci explique le peu d'intérêt rencontré par cet outil chez les utilisateurs.

IV.1.1.3.3 - Scalabilité Temporelle

Elle donne la possibilité de manipuler différents taux d'images (*framerate*) dans un flux vidéo unique. La couche de base, qui fournit l'image vidéo de base, peut être combinée avec la couche additionnelle pour atteindre un taux d'images supérieur. La couche additionnelle utilise l'information de la couche de base pour générer la vidéo finale.

Les applications possibles sont le support de décodeurs de générations différentes, ou encore l'utilisation dans des réseaux offrant plusieurs qualités de transmission

Dans ce cas-ci, le gain se passe après l'étape *DCT Inverse* et affecte principalement le processus de compensation de mouvement (voir Figure IV-2).

IV.1.1.3.4 - Scalabilité SNR (Signal to Noise Ratio)

Ce mode codage permet de transmettre un flux MPEG avec différents niveaux de qualité. Le flux de base, qui contient la vidéo codée de basse qualité, est transmis sur le canal avec un niveau de protection très élevé, et donc garantit à la réception une image quel que soit le taux d'erreur. Le flux additionnel apporte une meilleure qualité d'image, mais sera moins bien protégé dans la transmission. Ce système de codage permet donc une dégradation progressive de la qualité du signal vidéo décodé en fonction de la qualité de la transmission. La perte de données dans ce canal basse performance aura peu d'impact sur la qualité de l'image vidéo. Il faut remarquer que la couche de base et le flux additionnel ont la même résolution vidéo spatiale.

En pratique, un codeur SNR réalise les mêmes fonctions qu'un codeur standard, seul le processus de quantification est modifié. Les coefficients DCT sont d'abord quantifiés avec un pas de quantification élevé et transmis par codage VLC. Ensuite, l'erreur résiduelle due à la première quantification est quantifiée avec un pas de quantification bas et de la même façon transmise par codage VLC. Après quantification inverse et addition des deux valeurs, le coefficient DCT est injecté dans une boucle classique de compensation pour servir dans l'image de référence. Le décodeur SNR reproduit symétriquement quantification inverse et compensation de mouvement.

Le gain dans ce cas-ci se produit après le processus de quantification inverse. La couche additionnelle contient principalement des coefficients DCT, qui sont additionnés à ceux fournis par la couche de base. Cela permet de raffiner la qualité de l'image.

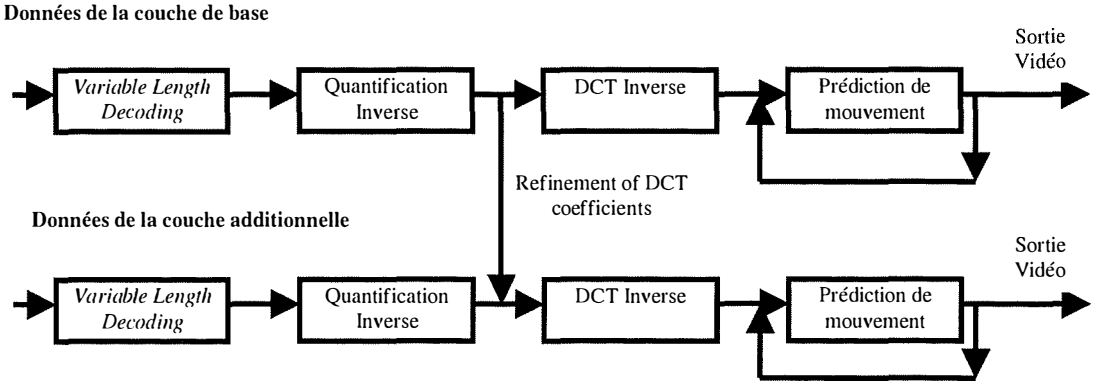


Figure IV-3 : Schéma de décodage avec la scalabilité temporelle

IV.1.1.3.5 - Partitionnement de données

Le partitionnement de données peut être utilisé pour diviser le flux de bits vidéo en plusieurs parties d'importance variable. Les éléments syntaxiques les plus importants sont transmis sur un canal haute performance, les éléments de moindre importance étant envoyés sur un canal moins performant. Par exemple, le canal haute performance transportera les éléments syntaxiques importants (l'en-tête) ainsi qu'un coefficient DCT, tandis que les autres coefficients seront envoyés sur le canal moins performant.

Un élément syntaxique spécial, appelé *Priority_Breakpoint*, est utilisé pour définir quels éléments du flux vidéo seront placés dans quelle partition.

Tableau IV-1 : Priority_Breakpoint

Valeurs du *Priority_Breakpoint*	Eléments de la syntaxe insérés dans la partition à haute priorité
1	En-tête de séquence, GOP, images, et slice jusqu'au extra_bit_slice dans le slice
2	Toutes les données du (1) + les données de macrobloc jusqu'à l'adresse d'incrément du macrobloc
3	Toutes les données du (2) + les données de macrobloc jusqu'au coded_bloc_pattern
64	Tous les éléments syntaxiques jusqu'au niveau du bloc, inclus le premier coefficient DCT
65	Toutes les données précédentes + 2 coefficients DCT
63+n	Toutes les données précédentes + n coefficients DCT
127	Toutes les données précédentes + 64 coefficients DCT

Un décodeur qui supporte le partitionnement de données doit d'abord décoder le flux de bits délivré par la partition 0, avant de switcher vers la partition 1.

Ce mode de codage est moins performant d'un point de vue de dégradation progressive du signal que le mode SNR. En effet, la suppression pure et simple des coefficients DCT entraîne des défauts de codage plus visibles qu'une surquantification. En revanche, le partitionnement de données a le mérite d'être beaucoup plus simple à mettre en œuvre.

IV.1.1.4 - Profils et Niveaux

Du fait du vaste éventail des applications censées être supportées par MPEG-2, la norme est devenue assez complexe. Cependant, une application peut ne pas avoir besoin de l'ensemble des caractéristiques de MPEG-2 Vidéo. De plus, l'équipement MPEG-2 aurait un coût prohibitif s'il devait supporter les spécifications complètes. Ainsi, la norme définit des profils et des niveaux qui caractérisent des sous-ensembles de MPEG-2 Vidéo.

Tableau IV-2 : Différents profils et niveaux

Profils	Niveaux
Simple Profile (SP)	*Low Level (LL)*
Main Profile (MP)	*Main Level (ML)*
SNR Scalable Profile (SNR)	*High 1440 Level (H14)*
Spatial Scalable Profile (Spatial)	*High Level (HL)*
High Profile (HP)	

IV.1.1.4.1 - Profil

Un profil est un sous-ensemble bien défini de la syntaxe vidéo. Certains éléments de la syntaxe MPEG-2 Vidéo ne sont pas valides et ne peuvent être décodés si le décodeur ne supporte qu'un profil bas. Par exemple, un profil *simple* ne supporte pas les *B-Picture*, de même que les profils *simple* et *main* ne supportent aucun type de scalabilité.

Tableau IV-3 : Caractéristiques selon les profils

Caractéristiques MPEG-2	*Simple Profile*	*Main Profile*	*SNR Profile*	*Spatial Profile*	*High Profile*
Format de chrominance	4 :2 :0	4 :2 :0	4 :2 :0	4 :2 :0	4 :2 :0 ou 4 :2 :2
Type d'image	I, P	I, P, B	I, P, B	I, P, B	I, P, B
Mode(s) de scalabilité	Aucun	Aucun	SNR	SNR ou spatial	SNR ou spatial

IV.1.1.4.2 - Niveau

Un niveau définit des valeurs pour certains paramètres dans le flux de bits vidéo. Il décrit notamment le nombre d'échantillons par lignes, le nombre de lignes par frame et le nombre de frames par secondes.

Tableau IV-4 : Caractéristiques selon les niveaux

Niveau	Résolution	Pixel/seconde	Débit maximum	Taille de mémoire tampon	Notes
Low	325 x 240 x 30	3.05 Millions	4 Mb/S	475.136 bits	CIF, équivalent au VHS
Main	704 x 480 x 30	10.40 Millions	15 Mb/s	1.835.008 bits	CCIR 601, studio TV
High 1440	1440 x 1152 x 30	40.00 Millions	60 Mb/s	7.340.032 bits	HDTV
High	1920 x 1080 x 30	62.70 Millions	80 Mb/s	9.781.248 bits	Production vidéo, standard SMPTE 240M

IV.1.1.4.3 - Profil@Niveau

Les profils et les niveaux sont combinés pour définir exactement quelle sélection ou quel sous-ensemble des outils MPEG-2 Vidéo est utilisé. Une combinaison très importante est ML@MP (*Main Level@Main Profil*). Cette combinaison définit un sous-ensemble des fonctionnalités de MPEG-2 Vidéo suffisant pour permettre la TV standard en broadcast (PAL ou NTSC). Les valeurs caractérisant cette combinaison sont présentées dans le Tableau IV-5.

Tableau IV-5 : Caractéristiques de la combinaison *Main_Level@Main_Profil*

Paramètres	
Echantillons/ligne	720
Lignes / image (frame)	576
Images (frames) / seconde	30
Echantillons de luminance/seconde	10 368 000
Taux max. pour les données vidéo (Mbits/s)	15
Taille maximale du buffer du décodeur (bits)	1 835 008

IV.1.1.4.4 - Compatibilité

Les profils et Niveaux sont organisés de façon hiérarchique et MPEG-2 Vidéo prescrit une compatibilité en arrière entre les différents profils et niveaux. En d'autres termes, un décodeur qui supporte un profil élevé doit aussi supporter les profils inférieurs.

IV.1.2 - MPEG-2 Partie 3 : AUDIO (ISO/IEC 13818-3)

IV.1.2.1 - Introduction

Parallèlement à ce qui a été expérimenté en étudiant les signaux vidéo, le signal audio résultant de la compression et de l'encodage est à la fois très puissant et très flexible. Il est à noter que comme pour la compression MPEG-2 Vidéo, MPEG ne définit pas de modèle d'encodage, il définit uniquement le format du flux de bits et le modèle de référence du décodeur.

La partie audio de la norme MPEG-2 est en grande partie basée sur la partie audio de MPEG-1, et le niveau de compatibilité est élevé (compatibilité dans les deux sens). Compatibilité en arrière d'abord: les équipements MPEG-1 existants peuvent décoder partiellement les signaux MPEG-2 en extrayant la partie compatible avec MPEG-1. Compatibilité en avant ensuite: les équipements MPEG-2 peuvent décoder les signaux MPEG-1.

IV.1.2.2 - Syntaxe MPEG-2 audio

IV.1.2.2.1 - Syntaxe

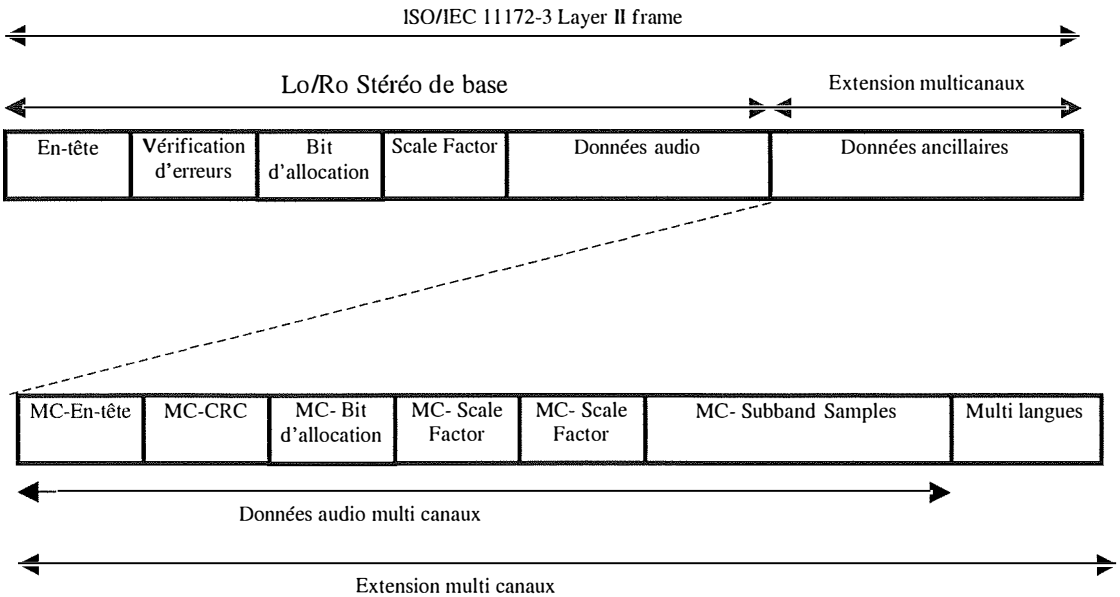


Figure IV-4 : Syntaxe audio

IV.1.2.2.2 - Détail de certains champs

L'*En-tête* est une structure commune aux 3 couches de MPEG-1 et de MPEG-2. Elle contient notamment un *ID* qui indique si le PDU est encodé selon MPEG-1 ou MPEG-2, un champ

Couche qui indique la couche utilisée (1, 2 ou 3), un champ *Mode* qui indique le mode utilisé (voir *2.4. Modes de codage*), et un champ *Copyright* qui indique si le flux est protégé par des droits ou pas.

Le champ *Vérification d'erreur*, codé sur 16 bits, est présent si le *bit de protection* dans l'en-tête vaut 0. Il est de type CRC (*Cyclic Redundancy Check*) et permet de détecter les erreurs dans le flux de bits. Les bits 16 à 31de l'en-tête sont toujours protégés dans les 3 couches.

Le champ des *Données audio* contient les échantillons audio. Un champ *Bit d'allocation* indique le nombre de bits utilisés pour représenter les échantillons de chaque canal. Il peut prendre une valeur entre 0 et 15. Il faut noter que pour les couches 2 et 3, la structure du PDU pour ce champ est différente, reflétant une compression et une technique de codage plus complexe.

Le champ des *Données Ancillaires* permet de transporter les informations d'extension multicanaux (disponible avec MPEG-2).

Pour un détail des champs, voir Annexes MPEG-2.

IV.1.2.3 - Modes de codage

Comme indiqué dans la structure du PDU, il y a plusieurs façons d'utiliser les capacités de transmission des canaux disponibles. Les 4 modes de codage utilisés dans MPEG-1 et MPEG-2 sont indiqués dans le Tableau IV-6.

Tableau IV-6 : Modes de codage

Mode de codage	Commentaire
Mono (Single)	Un signal monophonique est transmis.
Mono (Dual)	Deux signaux monophoniques indépendants sont transmis.
Stéréo	Un signal stéréophonique est transmis (canal audio gauche et canal audio droit transmis séparément).
Joint Stéréo	Un signal stéréophonique est transmis, mais les canaux d'information gauche et droit sont combinés au-dessus d'une certaine fréquence pour compresser le signal

IV.1.2.4 - Additions MPEG-2 Audio

MPEG-2 Audio, bien que reprenant les principaux éléments de MPEG-1 Audio, possède quelques caractéristiques supplémentaires qui sont présentées dans ce paragraphe.

IV.1.2.4.1 - Taux d'échantillonnage réduit de moitié

Avec MPEG-2, il est possible d'utiliser un taux d'échantillonnage réduit de moitié par rapport à celui de MPEG-1 et de conserver encore une très bonne qualité de son. C'est particulièrement intéressant par exemple pour les applications suivantes : les canaux de commentaires, les canaux multi langues et les canaux multimédia. Le bit ID est mis à 0 si ce taux d'échantillonnage est utilisé.

IV.1.2.4.2 - Extension multicanaux

IV.1.2.4.2.1 - Configuration surround

Pour permettre la transmission d'une représentation stéréophonique plus réaliste, MPEG-2 supporte 5 canaux audio, lesquels peuvent fournir ensemble une impression surround. Les 5 canaux sont le canal gauche (L), le canal droit (R), le canal centre (C), le canal surround arrière gauche (Ls) et le canal surround arrière droit (Rs). Cette disposition est aussi appelée *stéréo 3/2* car elle utilise 3 haut parleurs de face et 2 dans le dos.

Un canal *Low_Frequency_Enhancement* (LFE) est aussi disponible pour transmettre un signal *sub woofer* allant de 15 à 120 Hz. Ce canal est principalement utilisé pour les effets spéciaux.

IV.1.2.4.2.2 - Autres configurations

Le Tableau IV-7 montre les canaux que l'encodeur peut utiliser comme input.

Tableau IV-7 : Canaux en input

Combinaison #	# de canaux	Configuration	Canaux
1	5	3/2	L, R, C, Ls, Rs
2	5	3/0 + 2/0	L, R, C du prog #1, L2, R2 du prog #2
3	4	3/1	L, R, C et S (un seul canal surround)
4	4	2/2	L, R, Ls, Rs (pas de canal central)
5	4	2/0 + 2/0	L, R, L2, R2 (canal droit et gauche de 2 programmes différents)
6	3	3/0	L, R, C (pas de son surround)
7	3	2/1	L, R, S, (canal gauche et droit, plus un canal surround)
8	2	2/0	L, R (ou mode dual-channel)
9	1	1/0	Mo (canal mono unique)

Au niveau du décodeur, les canaux décrits ci-dessus peuvent être décodés et les combinaisons présentées dans le Tableau IV-8 peuvent être reproduites.

Tableau IV-8 : Canaux en output

Combinaison #	# de canaux	Configuration	Canaux de front	Canaux arrières
1	5	3/2	L, R, C	Ls, Rs
2	4	3/1	L, R, C	S (un canal surround)
3	4	2/2	L, R	Ls, Rs
4	3	2/1	L, R	S (un canal surround)
5	3	3/0	L, R, C	(pas de canal surround)
6	2	2/0	L, R	(pas de canal surround)
7	1	1/0	Mo (Mono)	(pas de canal surround)

Il est à noter que le canal (optionnel) LFE peut être utilisé avec n'importe quelle configuration. En outre, la spécification MPEG-2 permet jusqu'à 7 canaux multilingues ou de commentaires par programme.

IV.1.2.4.3 - Compatibilité et Matrixing

IV.1.2.4.3.1 - Principe

La compatibilité entre MPEG-1 Audio et MPEG-2 Audio a été un aspect crucial lors du développement de la partie audio de MPEG-2. Le but est d'accélérer l'acceptation par les utilisateurs en s'assurant que les produits basés sur MPEG-1 puissent fonctionner de manière satisfaisante avec des signaux MPEG-2, et vice versa. Cependant, il est évident qu'un décodeur MPEG-1 ne peut pas décoder les signaux à taux d'échantillonnage réduit, et il ne peut pas non plus offrir les nouvelles options, telles que LFE et le son surround, inclus dans le signal MPEG-2 encodé.
Le format de base des frames utilisées pour le codage en MPEG-2 Audio est le même que celui utilisé en MPEG-1. Le champ *Données Audio* transportant l'information des canaux L et R en MPEG-1 transporte en MPEG-2 un signal mixte compatible. Le champ des *Données Ancillaires* trouvé dans la structure des frames de MPEG-1 est utilisé pour transporter l'information d'extension multi-canaux de MPEG-2.

IV.1.2.4.3.2 - Cas du stéréo 3/2

On doit transmettre 5 canaux d'information audio pour utiliser le potentiel des décodeurs MPEG-2. En outre, les décodeurs MPEG-1 doivent pouvoir reproduire un signal de haute qualité. La solution est de mixer les 5 canaux d'information surround en 2 canaux que le décodeur MPEG-1 peut manipuler par défaut (comme canal d'information gauche et droit), et que MPEG-2 peut utiliser pour reconstruire le signal surround. Cela s'appelle le *matrixing* et le *dematrixing*. Ces 2 canaux universels sont appelés *Lo* et *Ro*, et sont construits dans l'encodeur MPEG-2. *Lo* et *Ro* sont définis comme suit :

$$\left.\begin{aligned} Lo &= L + a \,.\, C + b \,.\, Ls \\ Ro &= R + a \,.\, C + b \,.\, Rs \end{aligned}\right\}$$ Avec a et b des constantes décrivant l'importance accordée au canal central et au canal surround

Au total, on peut identifier 7 canaux audio différents (R, L, C, Ls, Rs, Lo, Ro). *Lo* et *Ro* sont par défaut transmis dans la partie compatible MPEG-1/2 des frames audio, et les 3 derniers canaux sont transmis dans le champ des *Données Ancillaires* de la manière indiquée dans le Tableau IV-9.

Tableau IV-9 : Canaux transmis dans le champ des données ancillaires

Combinaison #	Canal # 1	Canal # 2	Canal # 3
1	L	R	C
2	L	R	Ls
3	L	Rs	C
4	L	Rs	Ls
5	L	R	Rs
6	Ls	R	C
7	Ls	R	Rs
8	Ls	Rs	C

IV.1.2.4.4 - Adaptive Multichannel Prediction

Le son surround est diffusé à travers 5 canaux. Il existe souvent des redondances entre ces 5 canaux, à tel point que dans certains cas, le même morceau d'information apparaît dans 2 ou plusieurs des 5 canaux surround (avec un certain délai dans certains cas). La compression MPEG-2 audio peut utiliser cette redondance pour atteindre un meilleur taux de compression.

En pratique, les 3 canaux transportés dans le champ d'extension multicanaux peuvent être prédits à partir du couple *Lo/Ro* compatible MPEG-1.

Quand la prédiction est utilisée, l'encodeur transmet, au lieu du signal réel dans les canaux d'extension, les informations suivantes : un coefficient de prédiction, une erreur de prédiction et un délai de compensation. Avec ces informations, le décodeur peut reproduire le signal des canaux d'extension à partir des canaux *Lo* et *Ro*.

IV.1.3 - MPEG-2 Partie 1 : System (ISO/IEC 13818-1)

IV.1.3.1 - Introduction

Les parties audio et vidéo de la norme MPEG-2 définissent le format utilisé pour représenter l'information audio et vidéo. Cependant, afin d'utiliser ces données dans un chaîne complète de service vidéo, il faut ajouter des fonctionnalités supplémentaires. Ces fonctionnalités découlent à la fois des applications utilisant ces données vidéo et audio, mais aussi des technologies utilisées pour la transmission des données.

Prenons comme exemple une application de télédistribution standard. Plusieurs programmes doivent être acheminés chez le consommateur afin qu'il puisse choisir librement entre ces programmes. Cela signifie qu'à certains endroits, plusieurs flux d'audio et de vidéo doivent être multiplexés et amenés ensemble chez le consommateur. Ce multiplexage est généralement effectué à un endroit précis du réseau. Dans le cas d'un système de distribution par satellite, les différents programmes, amenés par les différentes stations de diffusion, sont multiplexés ensemble dans des stations satellites de lien. Cette collection de programmes, (aussi appelée bouquet) est alors transmise au satellite, qui l'envoie alors chez le consommateur (via un système *direct to the home broadcasting*).

Ce multiplexage doit en principe être une fonctionnalité du réseau sous-jacent (par exemple, si on utilise un réseau ATM, les différents canaux virtuels peuvent être utilisés pour les différents programmes). Cependant, cela entraîne une grande dépendance entre la technologie de réseau utilisée et les données transportées. Pour cette raison, l'ISO/IEC a développé sa propre spécification ; une spécification qui décrit comment les flux de bits audio et vidéo encodés doivent être multiplexés ensemble pour former les programmes réels. La spécification est indépendante de l'implémentation physique du réseau (et convient à la fois à des environnements sans et avec erreurs), et contient des informations permettant de décoder les flux de bits audio et vidéo appartenant à des programmes spécifiques.

IV.1.3.2 - Flux de transport / flux de programme

IV.1.3.2.1 - Deux types d'environnements

Typiquement, il y a deux façons d'amener les informations à l'utilisateur. De nos jours, pour regarder un film, cela peut se faire via *VCR* [*Video Cassette Recorder*] (dans ce cas, on utilise un système de distribution local basé sur un média, la cassette vidéo) ou via diffusion TV, dans ce cas le film est transmis à la TV par câble ou par satellite. Dans le cas de la diffusion TV, on utilise un système de distribution basé sur un réseau.
Le même concept, média ou réseau, peut être appliqué à une vidéo au format MPEG-2. On peut soit stocker la vidéo localement sur un disque dur ou un CD-ROM et la récupérer au

moment de la présentation, soit amener la vidéo par réseau et jouer les images vidéo en temps réel. La partie système de la norme MPEG-2 permet les deux procédés, qui ont bien sûr des exigences différentes en fonction de la technologie utilisée pour l'implémentation. MPEG-2 définit deux outils de base pour cela : le flux de transport et le flux de programme.

IV.1.3.2.2 - Flux de programme

Le flux de programme s'applique particulièrement aux médias sur disque dur et sur CD-ROM. Le flux de programme utilise des structures de données longues pour transporter les données audio et vidéo. Cela est possible uniquement dans les environnements à faible taux d'erreur, car une perte peut provoquer de gros problèmes au niveau de la qualité du transfert de l'information audio et vidéo. Cette partie n'est pas développée dans ce document.

IV.1.3.2.3 - Flux de transport

Le flux de transport est utilisé dans les environnements réseaux. Le flux de transport utilise des structures de données de tailles fixes relativement courtes qui peuvent être traitées facilement dans des environnements réseaux. Cela permet tout d'abord une optimisation des équipements réseaux qui traitent les paquets, et ensuite une petite partie seulement des informations est perdue si le paquet est corrompu pour quelque raison que ce soit.

IV.1.3.3 - Caractéristiques du flux de transport

IV.1.3.3.1 - Multiplexage

Une des caractéristiques les plus importantes du flux de transport est la capacité de multiplexer et démultiplexer plusieurs programmes.

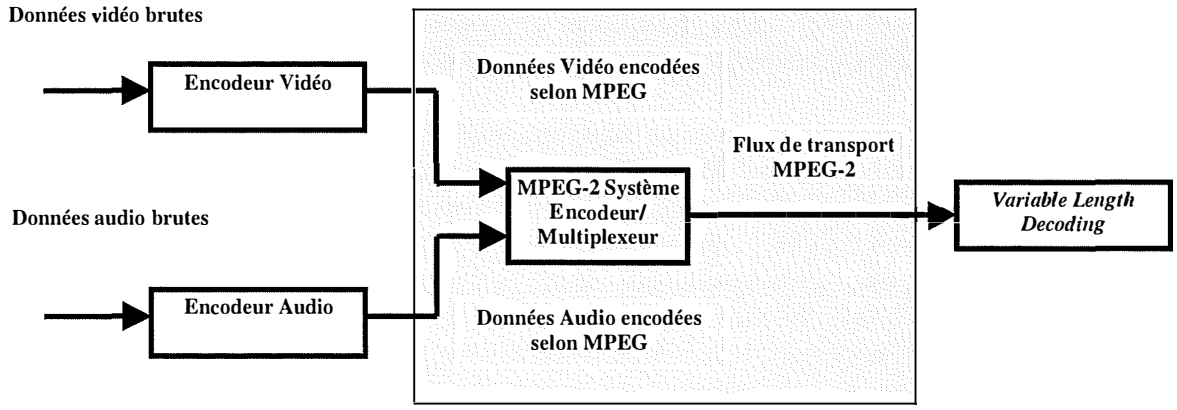


Figure IV-5 : Multiplexage

Un programme est lui-même constitué de plusieurs flux de bits audio et vidéo. La couche système de MPEG-2 possède les fonctionnalités nécessaires pour extraire un programme d'un flux de transport unique (qui contient une collection de programmes), pour extraire un sous-ensemble de programmes d'un flux de transport unique (qui contient une collection de programmes) et pour créer un flux de transport unique, contenant une collection de programmes à partir de plusieurs flux de transport.

IV.1.3.3.2 - Synchronisation

La synchronisation entre les flux audio et vidéo doit être conservée pendant le processus de multiplexage. Cela est possible en ajoutant un *timestamp* dans les éléments de données du flux de transport. (Voir IV.1.3.8 - Program Clock Reference)

IV.1.3.3.3 - Données privées

Les flux de transport qui sont multiplexés ensemble ne sont pas limités seulement aux flux audio et vidéo. MPEG-2 Système autorise des flux de bits définis par l'utilisateur, appelés flux de bits de données privées. On pourrait avoir par exemple des données de protocole réseaux (PDUs TCP/IP par exemple). Après avoir démultiplexé le flux de transport MPEG-2, les flux audio et vidéo seront décodés par le décodeur audio et vidéo, tandis que les données privées seront traitées par un logiciel adéquat.

IV.1.3.3.4 - Information de contrôle et de gestion

A côté de ces fonctions de multiplexage, l'encodeur/multiplexeur MPEG-2 ajoute aussi des informations de gestion aux différents flux audio et vidéo. (Voir IV.1.3.7 - Program Specific Information)
Certains en-têtes contiennent aussi des champs CRC, des bits de priorité, des indicateurs d'erreurs. Cependant, il n'y a pas de fonction de correction d'erreur. Ce type de fonctionnalité peut être pris en charge par les protocoles réseaux, qui peuvent utiliser les indicateurs fournis par la couche système de MPEG-2. (Voir IV.1.3.9 - Détection d'erreurs)

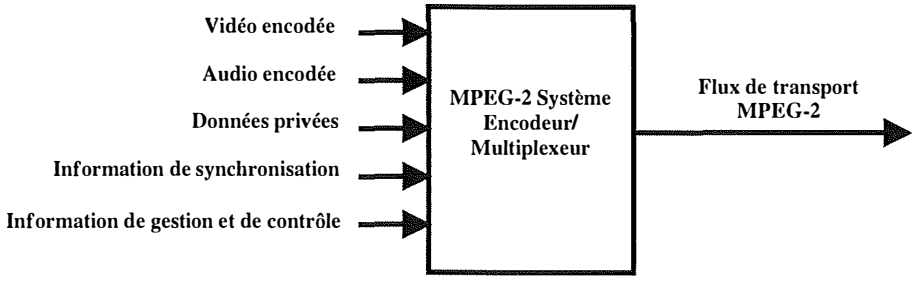
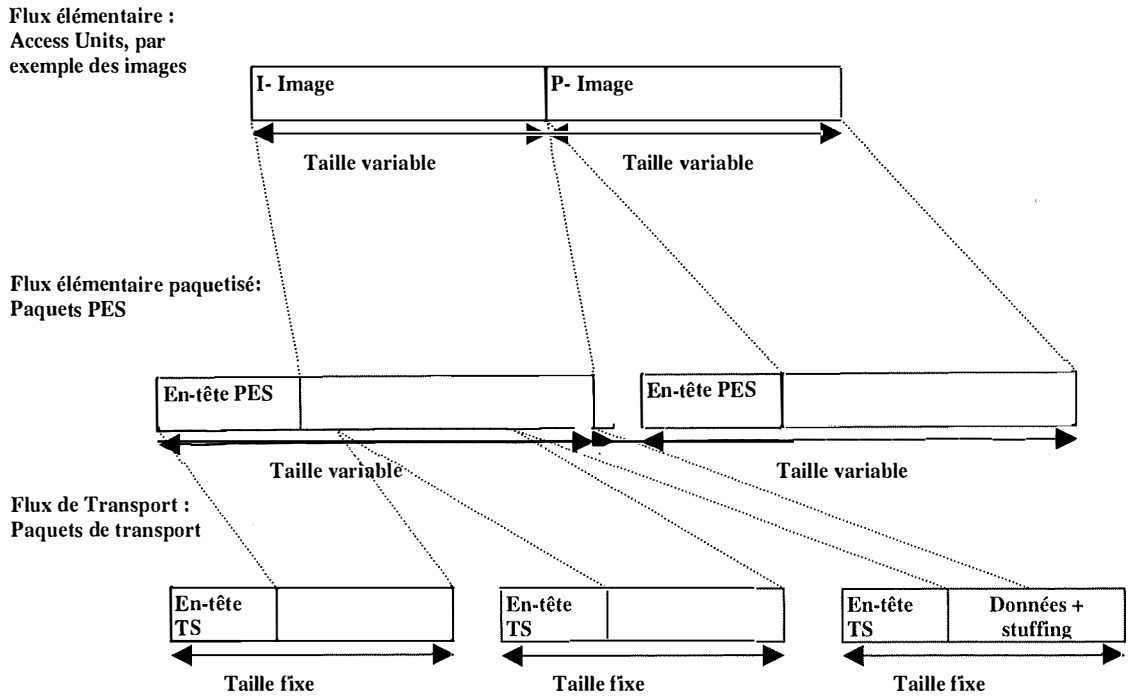


Figure IV-6 : Informations pouvant former un programme

IV.1.3.4 - Flux élémentaire – PES – Flux Transport

MPEG-2 Système utilise des structures de données que l'on appelle communément dans le monde de la communication de données des paquets. Les paquets sont toujours constitués d'un *en-tête* et d'un *contenu*, et peuvent être de taille fixe ou variable. L'idée de base derrière ce concept de paquet est de créer un mécanisme flexible pour transporter n'importe quel type de données. Généralement, l'*en-tête* contient les informations nécessaires pour traiter les données du *contenu* (par exemple le type d'image).

IV.1.3.4.1 - Relations entre les flux



IV.1.3.4.2 - Flux élémentaire

Dans MPEG-2, le flux de bits sortant d'un encodeur audio/vidéo ou le flux de bits de données privées est appelé flux élémentaire. Dans le cas d'audio ou de vidéo, ce flux élémentaire peut être organisé en *access unit*. Un *access unit* est, dans le cas d'un flux élémentaire de vidéo, une image, ou dans le cas d'un flux élémentaire audio, une frame audio.

IV.1.3.4.3 - PES

Un flux élémentaire est alors converti en flux élémentaire paquetisé, composé de paquets PES. Chaque paquet PES est composé d'un *contenu PES* (qui est une partie de taille variable du flux élémentaire) et d'un *en-tête PES*. En ayant la taille du *contenu* variable, le *contenu* d'un paquet PES peut contenir un *access unit* entier.

L'en-tête des paquets PES ajoute de l'information directement en relation avec le flux élémentaire (ex : audio/vidéo, copyrights…). Cette information est de manière générale indépendante des mécanismes utilisés pour acheminer l'information.

IV.1.3.4.4 - Flux de transport

Le paquet PES est alors inséré dans des paquets de flux de transport (TSP), composé d'un *en-tête* et d'un *contenu*. Des paquets de transport consécutifs forment un flux de transport MPEG-2.

Si les données du paquet PES ne remplissent pas complètement le paquet de transport, le paquet de transport est rempli avec du *stuffing* (du bourrage). Le début du prochain paquet PES est alors inséré dans le prochain paquet de transport. Cela permet au décodeur de facilement se synchroniser sur l'en-tête du PES, qui se trouve toujours au début du *contenu* du paquet de transport.

MPEG-2 Système distingue 2 types de flux de transport. Le premier est le flux de transport à programme unique (*Single Program Transport Stream* [SPTS]), qui contient différents flux

PES qui partagent tous une base de temps commune. Les différents flux PES peuvent transporter de la vidéo, de l'audio, voire même des informations sur les données, mais ils seront tous utilisés avec la même base de temps. Une application du SPTS est la transmission d'un film en plusieurs langues. Le second est le flux de transport à programmes multiples (*Multi Program Transport Stream* [MPTS]), qui multiplexe plusieurs SPTS.

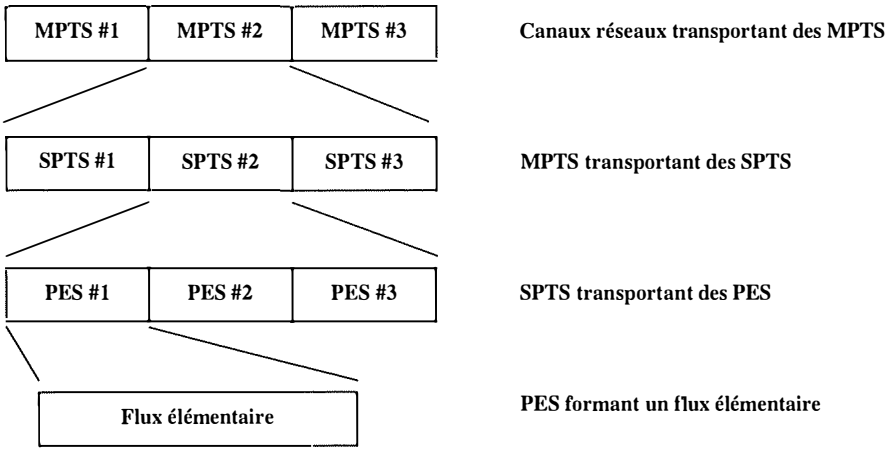


Figure IV-7 : Lien PES-SPTS-MPTS

IV.1.3.5 - Syntaxe MPEG-2 Système

IV.1.3.5.1 - Syntaxe hiérarchique

Comme pour MPEG-2 Vidéo, MPEG-2 Système utilise une syntaxe hiérarchique pour décrire les différents objets.

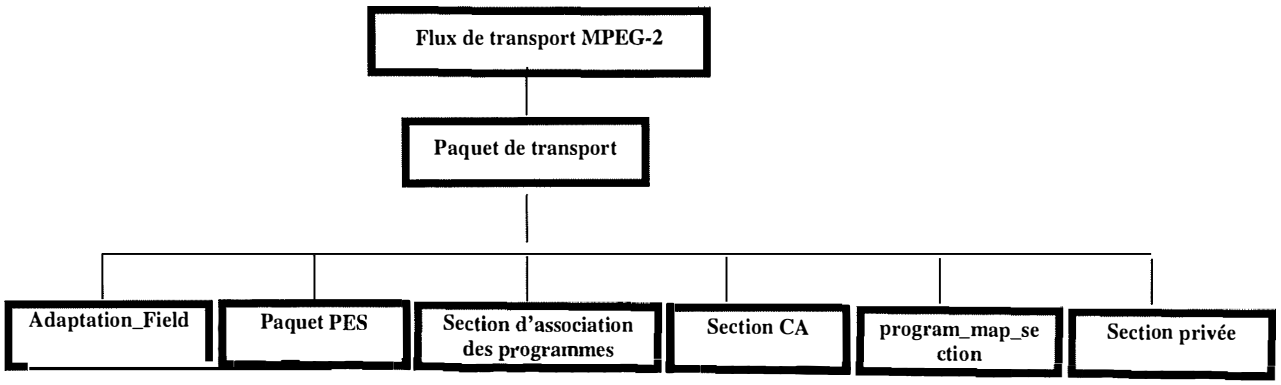


Figure IV-8 : Syntaxe système hiérarchique

MPEG-2 définit des paquets de transport de taille fixe d'une longueur de 188 bytes. Le paquet de transport MPEG-2 est constitué d'un *en-tête* de 4 bytes, d'un *Adaptation_Field* de taille variable et d'un *contenu* contenant le paquet PES.

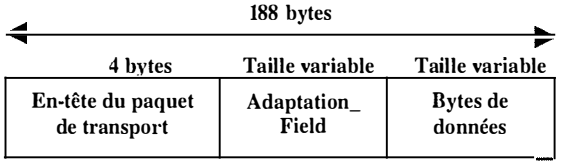


Figure IV-9 : Syntaxe paquet de transport

IV.1.3.5.2 - En-tête des paquets de transport

L'en-tête des paquets du flux de transport fournit des informations qui sont utilisées pour transporter et délivrer les flux. Cela inclut les outils permettant de multiplexer les différents flux d'information.

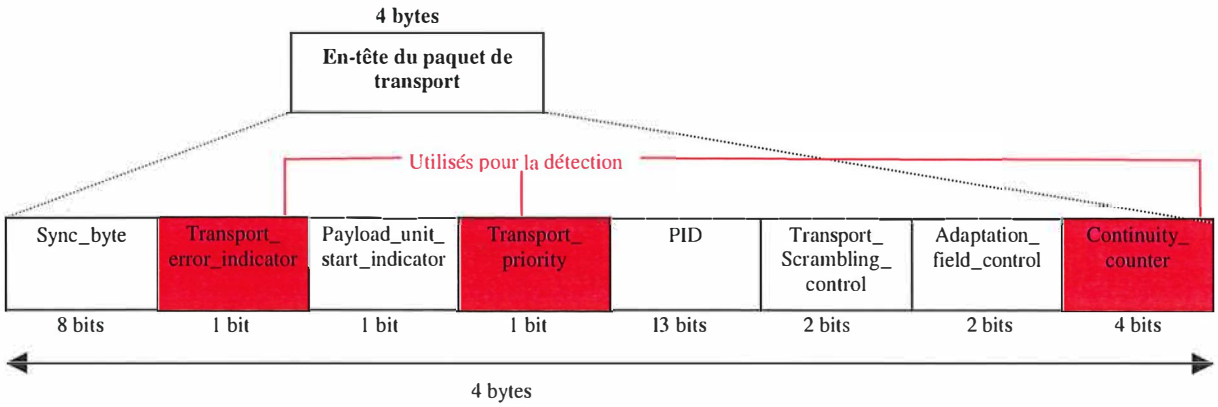


Figure IV-10 : Syntaxe de l'en-tête des paquets de transport

Le *PID* (*Packet IDentifier*) est un des champs les plus importants de l'en-tête. Le PID est utilisé non seulement pour identifier les paquets de transport qui contiennent des données PES provenant du même flux élémentaire, mais aussi pour définir le type de données qui sont transportées dans le *contenu* du paquet. Certaines valeurs de PID sont prédéfinies et ont une signification spéciale dans le contexte MPEG-2 Système. Certaines de ces valeurs sont illustrées dans le Tableau V-10.

Tableau IV-10 : Valeurs de PID

Valeur de PID	Description
0x0000	PAT
0x0001	CAT
0x0002 – 0x000F	Réservé
0x00010 à 0x1FFE	Disponible pour les flux PES, les *map tables*, les *network tables*

Par exemple, pour les paquets de transport qui ont un PID mis à 0, le *contenu* de ces paquets de transport contient une structure particulière appelée *tableau d'association de programmes* (PAT). Les paquets de transport avec un PID mis à 0x10 transportent des données PES en provenance d'un flux élémentaire audio ou vidéo.

IV.1.3.5.3 - Adaptation_Field

L'*Adaptation_Field* est un champ optionnel dans l'en-tête des paquets de transport qui contient des informations utilisées pour la gestion de l'horloge et pour les fonctions *raccordement* (voir IV.1.3.6 – Raccordement des flux de transport). Bien que les données qu'il contient soient très importantes pour le traitement des flux de transport MPEG-2, il n'est pas nécessaire dans tous les paquets de transport. De ce fait, ce champ a été déclaré optionnel et est utilisé sur demande dans les paquets de transport.

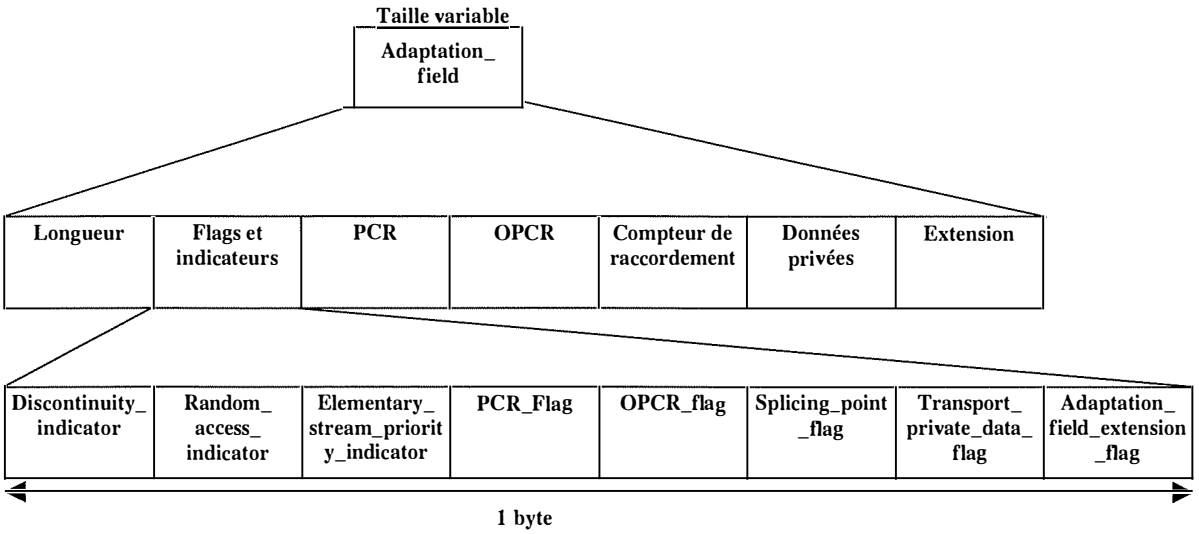


Figure IV-11 : Syntaxe de l'Adaptation Field

Le champ *PCR* (*Program_Clock_Reference*) est un des champs les plus importants de l' *Adaptation_Field*. Ce champ contient des timestamps qui sont utilisés par le décodeur pour synchroniser son horloge avec celle de l'encodeur.

L'*Adaptation_Field* contient un certain nombre de flags et d'indicateurs au début de la structure. Les flags déterminent le reste de la structure de l'*Adaptation_Field*. Les bits d'indication, quant à eux, sont utilisés pour donner de l'information au sujet du *contenu*. Par exemple, le bit *Elemntary_Stream_Priority* est activé si le *contenu* contient des données très importantes (une *I-picture* dans le cas d'un flux vidéo).

IV.1.3.5.4 - Paquet PES

Les paquets PES sont des paquets de taille variable et de format variable.

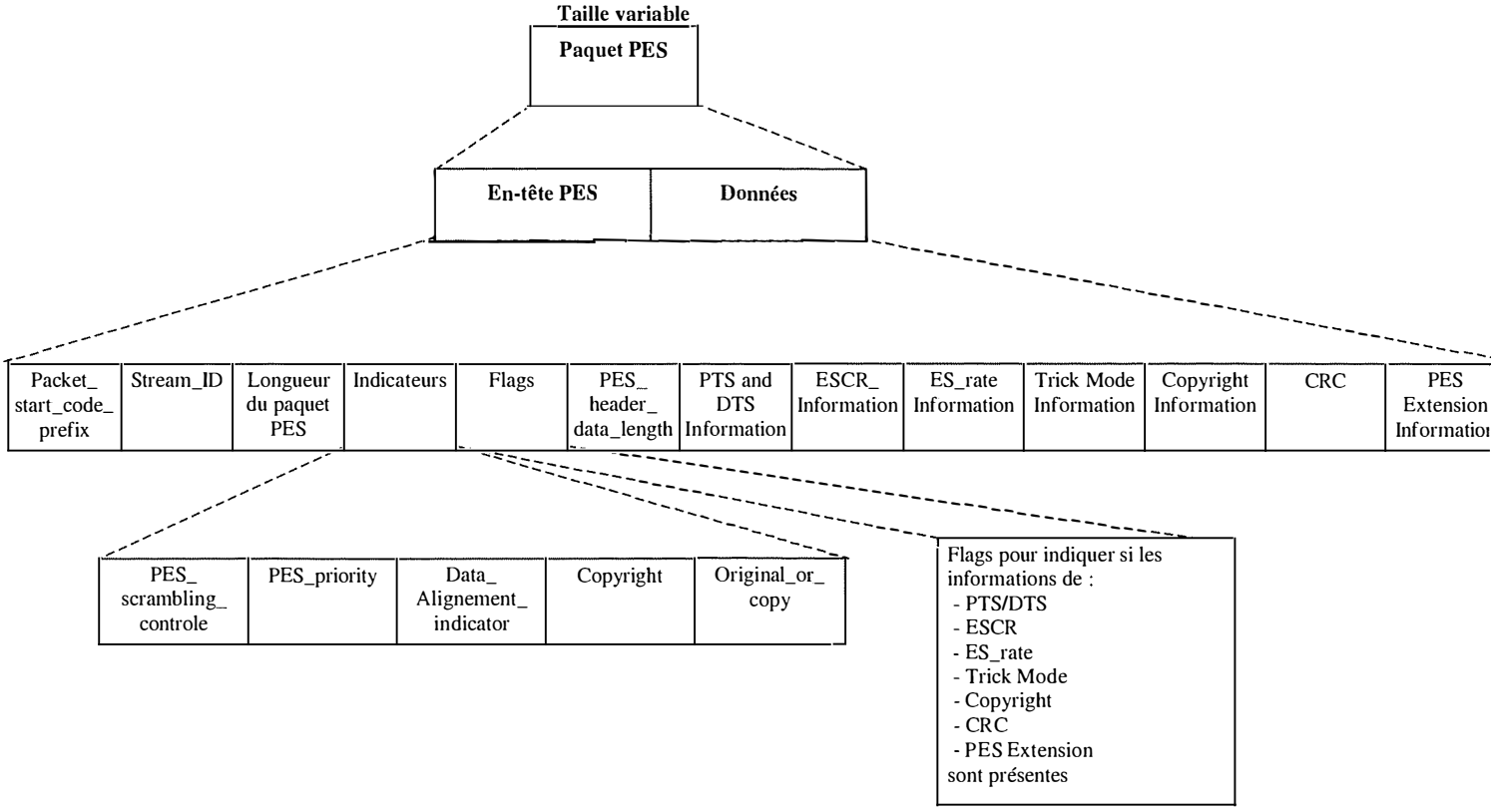


Figure IV-12 : Syntaxe des paquets PES

Le *Stream_ID* définit le format du paquet PES. Le Tableau IV-11 donne quelques exemples de valeurs de *Stream_ID.*

Tableau IV-11 : Valeurs de Stream_ID

Stream_ID	Description du flux
110x xxxx	Flux MPEG-2 Audio numéro xxxx, contenant des access units audio
1110 yyyy	Flux MPEG-2 Vidéo numéro yyyy, contenant des access units vidéo
1111 0010	Flux MPEG-2 DSM-CC, contenant des données de protocole DSM-CC

Aussi, on note notamment la présence de timestamps de présentation et de décodage, d'un bit de *Copyright* (ce bit est mis à 1, les données dans le paquet PES sont protégées par des droits) et d'une valeur de CRC (qui est calculée d'après le *contenu* des paquets PES précédents).

Des indicateurs sont utilisés pour donner de l'information additionnelle au sujet du contenu des paquets PES, parmi lesquels le *PES_Priorité*, qui permet d'indiquer la priorité du paquet PES.

IV.1.3.6 - Raccordement des flux de transport

L' *Adaptation_Field* et l'*Adaptation_Field_Extension* contiennent des éléments syntaxiques permettant de supporter la concaténation de deux flux PES différents. Un exemple de concaténation pourrait être l'insertion d'un flash de nouvelles dans le programme courant. La

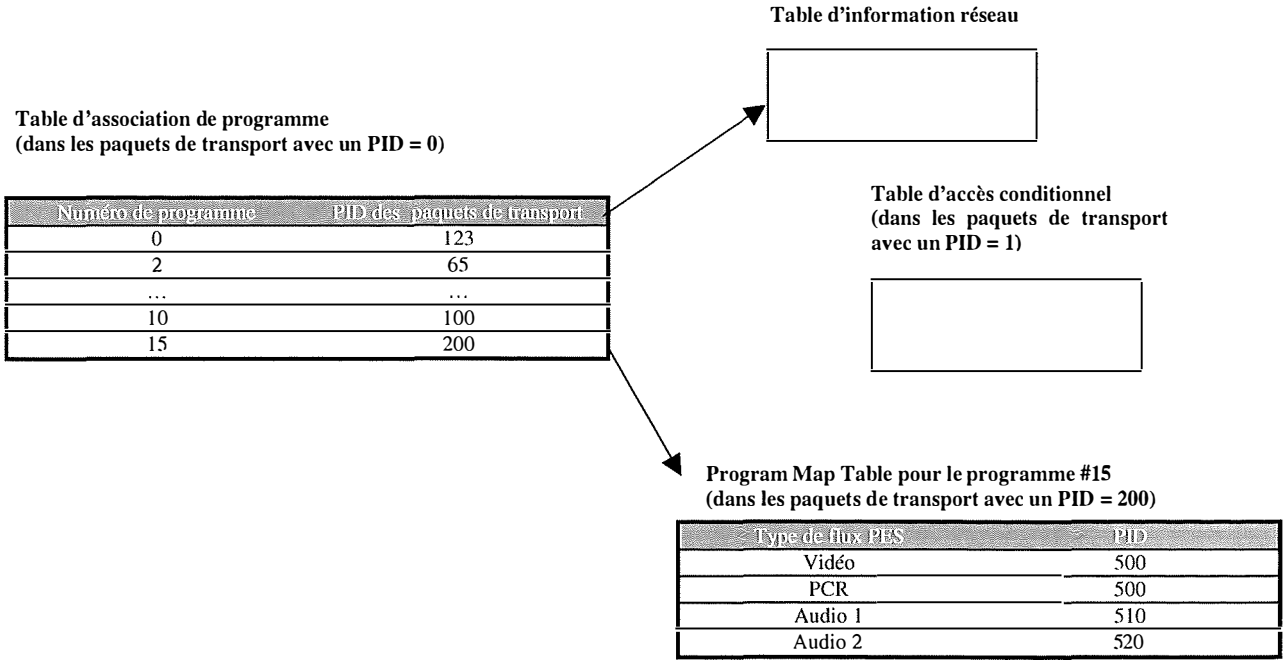
concaténation de deux flux PES nécessite la connaissance du début ou de la fin des *access unit* audio et vidéo. Pour éviter de décoder les paquets PES et les *access units* transportés, des *points de raccordement* sont indiqués par la syntaxe dans l'*Adaptation_Field* et l'*Adaptation_Field_Extension.* Les *points de raccordement* donnent des indications concernant l'endroit à partir duquel on peut insérer un nouveau programme dans le flux de transport courant. Ainsi, si le *compteur de raccordement* dans un paquet de transport atteint la valeur 0, cela indique que le dernier byte du *contenu* du paquet de transport actuel est aussi le dernier byte d'un *access unit* audio ou vidéo. A cet endroit, un nouvel *access unit* audio ou vidéo peut être inséré dans le flux PES actuel.

IV.1.3.7 - Program Specific Information

En plus des flux d'information audio et vidéo, MPEG-2 Système contient aussi des informations de gestion. Cette information est utilisée pour grouper différents flux vidéo et audio dans un programme. Selon la norme MPEG-2 Système, un programme est défini comme un certain nombre de flux élémentaires qui partagent une base de temps commune. Un exemple de programme pourrait être un flux vidéo, combiné avec deux flux audio et un flux de données privées. Les deux flux audio pourraient être utilisés pour transmettre plusieurs langues et le flux de données privées pourrait contenir des sous-titres.

Toutes les structures d'information destinées au contrôle et à la gestion des programmes sont groupées en *Program Specific Information* (PSI). PSI est constitué d'un ensemble de tables qui sont reliées ensemble.

IV.1.3.7.1 - Relations entre tables PSI (+ Exemple)



IV.1.3.7.2 - Table d'association de programmes (Program Association Table [PAT])

La PAT est le point de départ dans un flux de transport multiprogrammes. Elle se trouve dans les paquets de transport qui ont une valeur de PID égale à 0. Elle fournit l'information initiale sur l'identité des programmes contenus dans le flux de transport. Pour chaque programme du flux de transport, la PAT contient une entrée avec un numéro de programme et une valeur de

PID correspondante. Cette valeur de PID identifie les paquets de transport qui contiennent une autre table, la *Program Map Table*.

IV.1.3.7.3 - Program Map Table (PMT)

La PMT contient un champ appelé *Elementary_PID*. Ce champ fournit le PID de paquets de transport qui contiennent des paquets PES pour un programme spécifique. Un autre champ dans la PMT, le *Stream_type*, définit le type du flux PES trouvé dans les paquets PES identifiés par le champ *Elementary_PID*.

La PMT contient aussi ce qu'on appelle des *Stream_Descriptor*, qui sont utilisés pour donner de l'information supplémentaire sur les flux contenus dans le programme. Par exemple, un *Video_Descriptor* peut fournir des informations concernant le taux d'échantillonnage de la chrominance, la combinaison profil/niveau supportée, le framerate qui sont utilisés pour les flux vidéo élémentaires.

Tableau IV-12 : Descripteurs de flux

Nom du Descripteur	Description
Descripteur de flux vidéo	Fournit de l'information sur le flux vidéo codé (framerate, profil@niveau, format de chrominance)
Descripteur hiérarchique	Fournit l'information permettant de supporter la scalabilité vidéo
Data_stream_Alignement _Descriptor	Indique quel objet vidéo est au début du *contenu* du paquet PES (par exemple un slice, un *access unit* vidéo, un groupe d'images, une séquence vidéo)

IV.1.3.7.4 - Table d'information sur le réseau (Network Information Table [NIT])

Dans la PAT, le programme #0 a une signification spéciale. Le PID associé avec le programme #0 identifie les paquets transportant la NIT. La NIT transporte des informations concernant le réseau utilisé pour délivrer les flux de transport. MPEG-2 ne définit pas le contenu de la NIT, cette tâche est laissée au fournisseur du réseau.

IV.1.3.7.5 - Table d'accès conditionnel (Conditional Access Table [CAT])

Elle contient l'information relative aux méthodes de cryptage utilisées pour les données audio et vidéo. La CAT est transportée dans des paquets de transport avec une valeur de PID à 1.

IV.1.3.7.6 - PSI Table Sections

L'information des tables PSI peut être segmentée en sections, sections qui sont alors insérées dans des paquets de transport MPEG-2. L'en-tête des sections contient des champs indiquant le type de section, la longueur de section, le nombre réel de sections, et le nombre total de sections nécessaires à la construction de la table PSI. Les trous entre des sections consécutives ne sont pas permis et le dernier paquet de transport est rempli avec du bourrage.

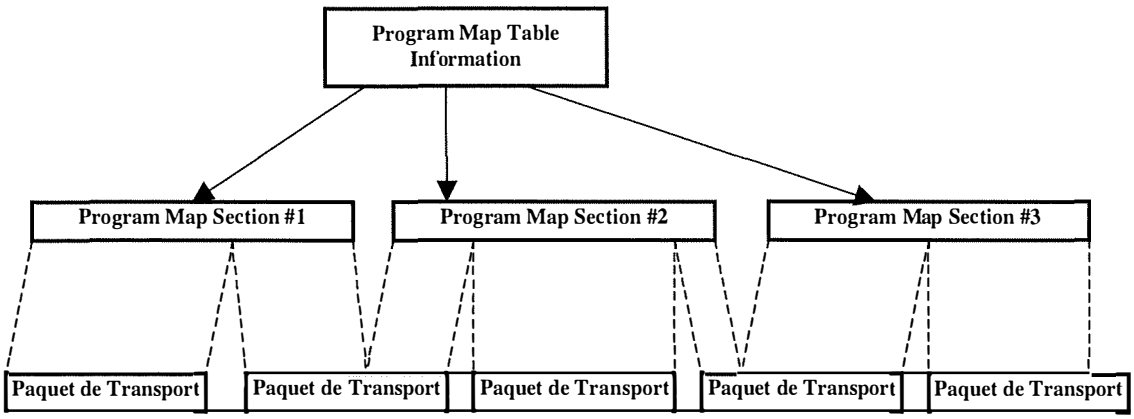


Figure IV-13 : Segmentation des tables PSI

IV.1.3.8 - Program Clock Reference

IV.1.3.8.1 - Principe

Durant le processus de décodage, un décodeur MPEG-2 collecte tous les paquets de transport possédant le même PID et construit/réassemble les *access units*. A ce stade, les données audio et vidéo ne sont pas encore décodées, et ne sont pas non plus présentées à l'utilisateur. Le moment où les *access units* seront réellement décodés et présentés est indiqué par les timestamps de décodage et de présentation (DTS et PTS).

Pour cela, le décodeur a besoin d'une horloge interne qui peut être utilisée pour déterminer le moment exact du décodage ou de la présentation. Cette horloge doit être très précisément synchronisée avec l'horloge qui a créé ces timestamps. Pour les flux de transport MPEG-2, cette horloge est appelée *Program_Clock* et peut être utilisée pour un ou plusieurs programmes du flux de transport MPEG-2. Pour s'assurer que la *Program_Clock* du décodeur est synchronisée avec l'horloge utilisée lors de l'encodage et du multiplexage des programmes, un timestamp PCR est transmis périodiquement.

Une fréquence de 27 MHz a été choisie pour le PCR afin de rester compatible avec le taux d'échantillonnage CCIR-601, qui est de 13.5 MHz pour les systèmes PAL et NTSC. Une fréquence de 27 MHz implique un incrément de l'horloge tous les 37 ns, qui nécessite un compteur de 42 bits pour couvrir les 24 heures d'une journée.

Le PCR est transporté en deux parties dans l'*Adaptation_Field* de l'en-tête des paquets de transport, à savoir dans le *Program_Clock_Reference_Base* et dans le *Program_Clock _Reference_Extension*. Les deux parties représentent 2 compteurs, tournant respectivement à 90 KHz et 27 MHz. Dès que le compteur tournant à 27 MHz atteint la valeur 300, il est remis à 0 et le compteur tournant à 90 KHz est incrémenté d'une unité. Pourquoi deux compteurs ? D'abord parce que MPEG-1 utilise une base de temps de 90 KHz. Afin de rester compatible, la partie à 27 MHz a été introduite dans un champ d'extension. Ensuite pour rester compatible avec le format des timestamps de décodage et de présentation qui utilisent aussi l'horloge système comme référence. Ils font 33 bits et peuvent être facilement comparés avec la valeur du champ de base PCR.

Program_Clock _Reference_Base (33 bits)	Program_Clock _Reference_Extension (9 bits)

Pour s'assurer que l'information de temps est toujours présente dans le système, MPEG-2 Système spécifie certaines obligations au niveau de la fréquence de codage des différentes timestamps. Ces contraintes sont indiquées dans le Tableau IV-13.

Tableau IV-13 : Fréquence des timestamps

Timestamp	Transmis au mois tout les :
Program Clock Reference	0,1 seconde
Presentation Timestamp	0,7 seconde
Decode Timestamp	Optionnel

La PMT d'un programme définit dans quels paquets de transport on peut trouver les timestamps pour ce programme en spécifiant les valeurs de PID de ces paquets de transport. L'horloge du décodeur est initialisée par le premier PCR transmis. Quand le timestamp suivant arrive (au moins à des intervalles de 0.1 seconde), la valeur actuelle de l'horloge du décodeur doit être exactement la valeur de ce timestamp. Sinon, l'horloge du décodeur doit être ajustée en se basant sur la différence entre l'horloge interne et la valeur de PCR.

IV.1.3.8.2 - Délai réseau

Le fait d'utiliser un réseau pour transmettre des timestamps de synchronisation engendre quelques problèmes. Cela suppose que le délai de transmission entre l'émetteur et le receveur est constant. S'il y a une variation dans le délai, l'horloge du décodeur en sera affectée.

Ce problème est particulièrement important si les données sont transmises sur un réseau qui introduit des délais.

IV.1.3.8.3 - Délai dû au multiplexage

Même si le réseau garantit un délai constant, un délai peut être introduit par le traitement de la couche MPEG-2. Les multiplexeurs des flux de transport MPEG-2, qui influencent la couche système de MPEG-2 directement, ajustent la valeur du PCR en fonction du délai qu'ils introduisent.

IV.1.3.9 - Détection d'erreurs et priorité dans MPEG-2 Système

Les mécanismes de détection d'erreurs s'appuient sur des champs contenus pour la plupart dans l'en-tête des paquets de transport. Premièrement, le *Transport Error Detection Bit* peut être activé par les équipements MPEG-2 (par exemple par un multiplexeur MPEG-2 de flux de transport) pour indiquer que le paquet de transport comporte une erreur dans l'en-tête ou dans le *contenu*. Deuxièmement, le *Continuity Counter Bits* (codé sur 4 bits) est incrémenté à chaque paquet de transport possédant le même PID.

L'en-tête contient également un *Transport Priority Bit* qui est utilisé pour indiquer un paquet de haute priorité. Ce paquet doit être traité de façon prioritaire si le réseau est congestionné. Un *Elementary Stream Priority Bit*, dont le rôle est similaire au *Transport Priority Bit*, est aussi présent au niveau des paquets PES.

A côté de ces bits d'indication, MPEG-2 utilise aussi des *checksums* (CRC) à différents endroits pour protéger l'information transportée. L'en-tête des paquets PES peut contenir un champ optionnel, qui est utilisé pour transmettre une valeur CRC calculée sur les bytes de données du paquet PES précédent. Des valeurs de CRC sont aussi utilisées pour chacune des tables PSI.

IV.1.4 - MPEG-2 Partie 6 : DSM-CC (ISO/IEC 13818-6)

Cette partie a été finalement approuvée comme norme internationale en juillet 1996.

IV.1.4.1 - Introduction

Le déploiement des services à large bande chez les utilisateurs finaux dépend de la disponibilité de protocoles ouverts. Sans de tels protocoles, chaque service demanderait sa propre interface pour être accessible.

Dès lors, à côté des aspects audio et vidéo des systèmes vidéo numériques, le groupe de normalisation MPEG s'est également penché sur ces domaines plus orientés réseaux. Pour ce faire, ils ont défini le protocole DSM-CC, qui est lui-même constitué d'un ensemble de protocoles.

IV.1.4.2 - Relation avec les autres protocoles

Le protocole DSM-CC complète les autres protocoles réseaux, tels que les protocoles de transport, de façon à satisfaire les exigences des applications vidéo.

DSM-CC est une couche indépendante. De cette façon, une application qui utilise DSM-CC ne doit pas s'occuper de la couche de transport sous-jacente entre le serveur et le client. Cela signifie qu'une même application peut être distribuée au travers une multitude de réseaux.

L'aspect crucial de DSM-CC est sa flexibilité. Chaque protocole de DSM-CC peut être utilisé seul, ou avec d'autres protocoles, selon l'application considérée. Par exemple, le DVB (*Digital Video Braodcasting*, voir *Applications MPEG-2*) utilise un sous-ensemble des messages DSM-CC U-N et U-U.

IV.1.4.3 - Modèle de référence

DSM-CC a défini un modèle simple de référence. Le modèle est composé d'un serveur et d'un client (appelés tous deux utilisateurs) qui utilisent un réseau pour communiquer entre eux.

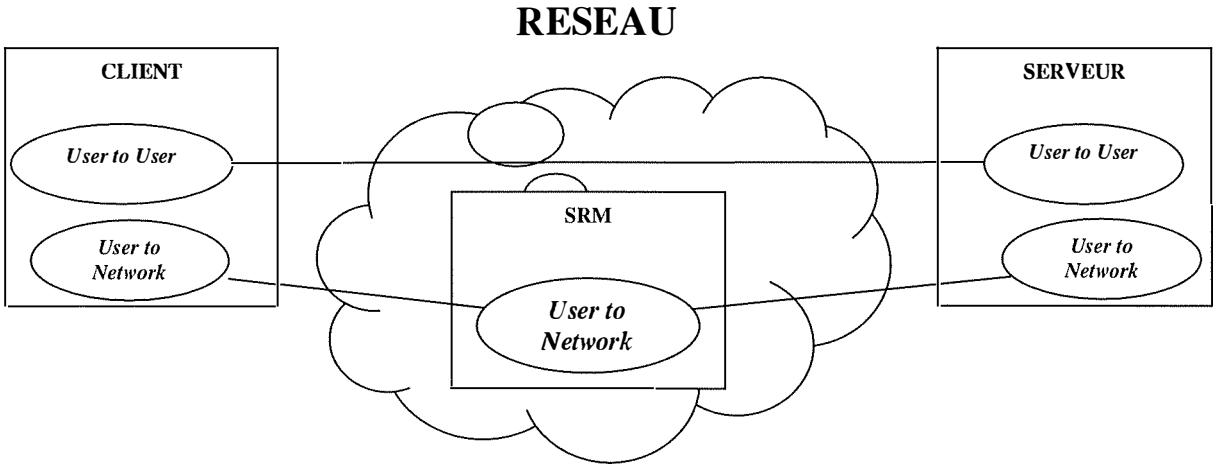


Figure IV-14 : Modèle de référence

La définition de réseau est large : c'est une collection d'éléments qui communiquent entre eux et qui permettent de mettre en connexion des utilisateurs.
La définition de connexion est également large : c'est une capacité de transport qui permet le transfert d'informations entre plusieurs utilisateurs finaux.
Le but de MPEG est que le protocole DSM-CC soit applicable à travers différents réseaux physiques avec des réalisations de connexions différentes.

IV.1.4.3.1 - Opérations U-N (User-to-Network)

Le rôle du protocole DSM-CC *User-to-network* est d'établir une session entre le réseau et l'équipement de l'utilisateur.

Une session est un concept clé de DSM-CC. Une session est définie comme une association entre 2 utilisateurs, qui permet de grouper les ressources nécessaires pour l'instanciation d'un service. Un client accède à un service en établissant une session avec un serveur. A la fin du service, la session est terminée.

Les messages U-N sont échangés sur des connexions U-N, leur but étant de contrôler les sessions et les ressources réseau. Les messages ne doivent pas être transportés de façon fiable, mais les messages corrompus doivent être détectés et jetés. Le service de transport doit être capable de délivrer les messages U-N en entier (les couches inférieures pouvant faire de la segmentation et du réassemblage), mais il ne doit pas les délivrer nécessairement dans le bon ordre. Les protocoles tels que UDP au-dessus de IP ou AAL5 au-dessus de ATM satisfont à ces exigences.

IV.1.4.3.2 - Opérations U-U (User-to-User)

Un flux d'information *User-to-User* (une connexion U-U) est utilisé entre un client et un serveur. En général, le protocole utilisé pour transporter les informations à travers ces connexions n'est pas spécifié par DSM-CC, c'est une question d'entente entre le client et le serveur. Cependant, DSM-CC définit un ensemble de services génériques qu'un serveur doit fournir à un client. L'appel à ces services se fait via le protocole RPC au travers d'une connexion U-U.

Une session DSM-CC comprend généralement plusieurs connexions U-U. Les sessions sont représentées comme étant un flux de contrôle, pour les messages RPC, et un flux MPEG-2. Les applications sophistiquées utilisent plusieurs connexions, par exemple une connexion pour transporter la vidéo montrée dans une fenêtre et une connexion rapide pour l'arrière-plan. Un client peut recevoir des informations de plusieurs sources pendant une session.

DSM-CC permet la distinction entre la gestion des sessions et des ressources, et les protocoles de contrôle des connexions. Un réseau peut utiliser une session DSM-CC et le protocole de gestion des ressources mais aussi implémenter le contrôle de la connexion avec les protocoles du réseau de transport sous-jacent.

IV.1.4.3.3 - Différences

Ces deux interfaces DSM-CC sont fondamentalement différentes. L'interface *User-to-network* a plus de points communs avec un protocole de signalement de type *OSI Layer 3* (définit des structures de PDU et des procédures pour l'établissement et la clôture d'une session). La partie *User-to-user* est plus orientée application, et utilise une approche orientée objet.

IV.1.4.3.4 - SRM

Le SRM (*Session and Ressource Manager*) a plusieurs rôles. C'est tout d'abord une entité qui régule les connexions serveur/client selon une politique établie à la souscription du service. Il est ensuite l'entité qui fournit les informations de configuration à l'utilisateur. Il permet enfin l'authentification du client.

IV.1.4.4 - Opérations User-to-network

Les opérations *User-to-network* se focalisent principalement sur le contrôle et la gestion des sessions entre les équipements de l'utilisateur (serveur ou client) et le réseau. Pour supporter ces opérations *User-to-network*, DSM-CC définit des messages et des séquences de commandes.

IV.1.4.4.1 - Messages

Un message DSM-CC *User-to-network* se compose toujours d'un en-tête et d'un *contenu.*

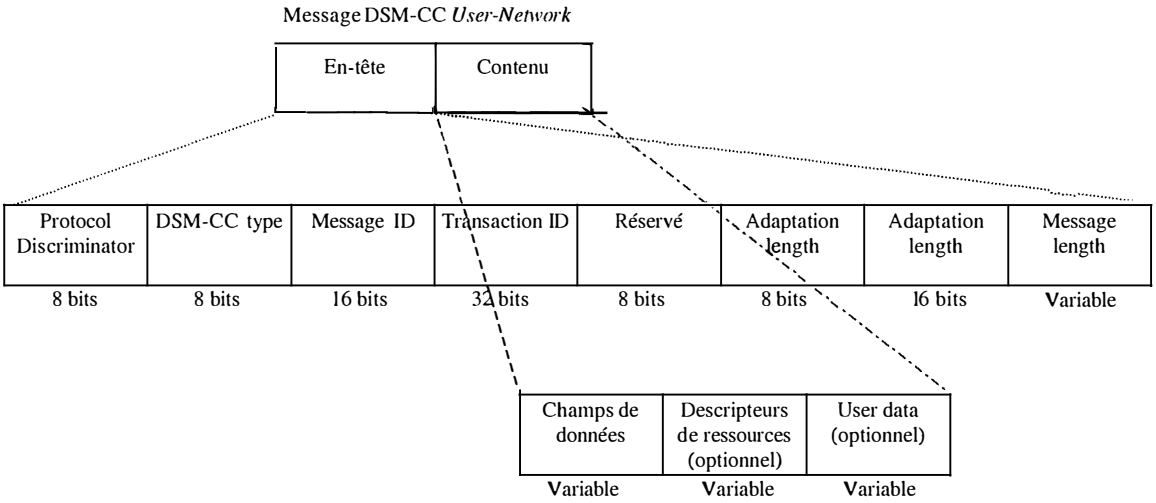


Figure IV-15 : Syntaxe des messages U-N

IV.1.4.4.1.1 - En-tête

Le champ *Dsmcc_Type* identifie le type de message DSM-CC. DSM-CC distingue deux types de messages pour l'interface *User-to-network*. Premièrement, il y a les messages de configuration *User-to-network*. Un client utilise ces messages pour se configurer par rapport au réseau auquel il est attaché. Cette configuration peut être initiée par l'utilisateur (messages *UNConfigRequest*) ou par le réseau (messages *UNConfigIndication*). Après une séquence de messages *UN Config*, le client est au courant des paramètres spécifiques du réseau, tels que la façon dont les identificateurs de session sont alloués, ou la façon de communiquer avec le SRM (adresse IP par exemple). La partie *UN Config* de DSM-CC est un protocole indépendant. N'importe quelle application qui requiert une configuration initiale avec le réseau peut utiliser cette partie de DSM-CC.

Tableau IV-14 : Types de messages

Nom du message	Description
UNConfigRequest	Envoyé de l'utilisateur vers le réseau pour configuration
UNConfigConfirm	Envoyé du réseau vers l'utilisateur en réponse à un UNConfigRequest

UNConfigIndication	Envoyé du réseau à l'utilisateur pour configurer un appareil utilisateur
UNConfigResponse	Envoyé de l'utilisateur au réseau en réponse à un UNConfigIndication

Deuxièmement, il y a les messages de gestion des sessions et des ressources *User-to-network*. Ce groupe de messages est utilisé pour établir et gérer les sessions applicatives vidéo. Il constitue une part essentielle la spécification DSM-CC.

Le *messageID* contient 3 parties : le *discriminateur du message*, le *scénario de message* et le *type de message*. Le premier indique si le message est passé entre un client et le réseau ou entre un serveur et le réseau. Le deuxième décrit le scénario dans lequel le message est utilisé (par exemple pendant l'établissement d'une session ou la clôture d'une session). Le troisième indique si l'utilisateur ou le réseau envoie le message et si le message échangé est initié par l'utilisateur ou le réseau. Il peut être un *Request* (utilisateur vers réseau), un *Confirm* (réseau vers utilisateur), ou un *Response* (utilisateur vers réseau en réponse à un *Indication*).

IV.1.4.4.1.2 - Contenu

Le format du *contenu* dépend du *messageID* dans le type de message DSM-CC. Le *contenu* contient des champs de données et des descripteurs de ressource.

Les champs de données ont un nombre fixe de bytes et contiennent des valeurs simples, à savoir un compteur, un identificateur de session et un identificateur de client. Comme DSM-CC opère dans un environnement où les ressources réseau sont limitées et onéreuses, les ressources sont acquises si nécessaires et relâchées dès que le service est terminé. De ce fait, l'identificateur de session est un champ très important. Il identifie de manière unique cette session dans le réseau. Toutes les ressources appartenant à une instance de cette session sont marquées de cet identificateur. Les implémentations au niveau du client et du serveur peuvent utiliser cet identificateur de session pour garder une trace des ressources appartenant à une session. Lors de la clôture de la session, les ressources identifiées par l'identificateur de session seront relâchées. L'identificateur de session est aussi utilisé si certaines ressources doivent être ajoutées ou relâchées lors d'une session spécifique. Il se révèle également utile pour la facturation et l'administration du réseau, ainsi que pour disposer des ressources d'une session lorsque celle-ci est terminée.

L'ajout et la suppression des ressources associées à une session se font avec les messages *AddRessource Message Set* et *DeleteRessource Message Set.* Dans ces messages, le serveur décrit les ressources à l'aide de descripteurs de ressources. Un descripteur de ressources contient des informations telle que la bande passante nécessaire pour des données vidéo.

IV.1.4.4.2 - Séquences de commandes

Les messages sont utilisés en séquences de commandes. Il existe 3 types de séquences de commandes.

IV.1.4.4.2.1 - Séquences de commandes initiées par l'utilisateur

L'utilisateur peut tout d'abord utiliser une séquence de commandes pour établir une session. La séquence de commandes est traitée quand un client est on-line et qu'une session applicative doit être établie avec le serveur.

L'utilisateur peut également initier une séquence de relâchement d'une session lorsqu'il quitte l'application. Dans ce cas, les ressources précédemment allouées peuvent être relâchées.

Le client peut aussi initier une requête de statut afin d'obtenir des informations concernant les sessions actives à cet instant, la configuration utilisée pour une session, et le statut d'une session spécifique.

IV.1.4.4.2.2 - Séquences de commandes initiées par le serveur

Le serveur peut tout d'abord initier une séquence de commandes pour établir une session. En effet certaines applications, comme les applications de téléapprentissage, requièrent que la session soit établie à partir du serveur.

Le serveur peut aussi demander l'ajout de ressources.

Le serveur doit enfin dans certains cas pouvoir fermer une session, pour, par exemple, des besoins de maintenance du système.

IV.1.4.4.2.3 - Séquences de commandes initiées par le réseau

Le réseau peut initier une séquence de terminaison de session, si, pour quelque raison, le réseau n'est plus capable de fournir le service.

Il doit pouvoir aussi interroger les équipements de l'utilisateur par des requêtes client ou serveur. Le résultat de cette interrogation peut mener à la fin de la session initiée entre le serveur et le réseau, si le client ne répond plus aux requêtes de statut.

IV.1.4.5 - Opérations User-to-user

Cette partie de DSM-CC s'occupe de la communication de bout en bout entre un serveur et un client. Ce type de communication est transparent pour le réseau. Il existe 2 types d'opérations.

IV.1.4.5.1 - Application Download Communication

Ces opérations sont principalement utilisées pour le chargement de code exécutable du serveur vers le client. Le protocole de chargement de DSM-CC est un protocole léger et rapide permettant de charger des données ou des logiciels, d'un serveur vers un client. Quand un client établit une session avec un serveur, DSM-CC permet au serveur de charger vers le client un environnement opérationnel complet.

Afin de s'assurer que les données chargées vers un matériel utilisateur fonctionneront effectivement sur ce matériel, des informations de compatibilité doivent être envoyé par le serveur avant que le chargement commence. DSM-CC fournit des descripteurs de compatibilité qui permettent à un matériel de se décrire à un serveur.

Il existe 3 types de chargement :

- Dans un scénario de service sur demande, cela pourrait être, par exemple, un logiciel applicatif de navigation qui est chargé directement après que la session entre le client et le serveur ait été établie.

- Pour une communication de chargement standard, DSM-CC définit un protocole basé sur des messages simples, qui implémente aussi un mécanisme de base de contrôle de flux de données.

- A côté de ces protocoles simples, DSM-CC a prévu l'utilisation d'une approche broadcast à des fins de chargement. DSM-CC a introduit l'idée d'un carrousel de données, où les données sont continuellement fournies sur un canal de chargement bien défini. Les clients peuvent se brancher sur ce canal, identifier les données offertes pour ce chargement en analysant périodiquement les messages de contrôle de chargement transmis, et finalement capturer les données qui les intéressent.

IV.1.4.5.2 - Communication Client-Serveur

Après qu'une session ait été établie entre le client et le serveur, l'application logicielle réelle implémentant le service peut être démarrée. Ce logiciel consiste typiquement en 2 composants : un, exécuté chez le client et l'autre, exécuté par le serveur. Les deux composants communiquent ensemble pour réaliser certaines tâches. Le logiciel client fournit une interface utilisateur, permettant à l'utilisateur de naviguer et d'utiliser le service à proprement parler. Au niveau du logiciel serveur, les requêtes du client sont traitées (par exemple, un film est démarré). Cette communication est très orientée application et est la plupart du temps transparente pour le réseau.

IV.1.4.5.2.1 - Corba

CORBA est une architecture orientée objet qui permet de supporter des applications distribuées. Cette architecture permet de définir des interfaces qui sont utilisées pour accéder aux fonctions et services implémentés par les objets distribués. Les interfaces sont décrites en utilisant le langage IDL (*Interface Description Language*). DSM-CC utilise en grande partie les concepts CORBA pour définir les interfaces des différentes fonctions qui sont censées être fournies par un serveur. Cette compatibilité avec CORBA a ses avantages et ses inconvénients. D'une part, c'est bien sûr un grand avantage de supporter une plate-forme très largement utilisée. Cela devrait être assez facile d'avoir des composants pour le service vidéo dans des plus grands systèmes d'applications logicielles. D'autre part, CORBA requiert pour le moment des systèmes performants (à la fois en terme de mémoire et de puissance de traitement).

IV.1.4.5.2.2 - Interface VCR

La première interface proposée est l'interface VCR. Les flux MPEG-2 contiennent leur propre horloge interne de façon à pouvoir synchroniser l'audio et la vidéo. Cependant, pour réaliser les fonctionnalités VCR dans une application de vidéo sur demande interactive, c'est-à-dire afin de permettre un positionnement aléatoire à l'intérieur du flux et différentes vitesses de marche, on a besoin de NPT (*Normal Play Time*). NPT avance normalement en mode *play*, avance rapidement quand l'utilisateur sélectionne l'option avance rapide pour un flux, ou décroît rapidement quand le rembobinage est sélectionné. Ces commandes doivent être transmises du client au serveur. Elles peuvent être implémentées en utilisant la partie *User-to-user* de DSM-CC.

Un scénario typique de vidéo sur demande est expliqué ici. Il reprend des concepts introduits précédemment.

1) Au début, le client (par exemple un *set top box*) initie un *U-N Config* pour se configurer en utilisant l'information stockée sur le réseau.
2) Un *U-N Session Setup* est utilisé pour établir une session entre le client et le serveur à travers le réseau.
3) Les connexions pour le chargement sont établies, en tenant compte des informations sur le contexte reçues par le client dans les messages attachés à la session U-U.
4) Les logiciels sont ensuite chargés par le client. En utilisant les fonctions du *U-U Directory*, un menu d'option est offert à l'utilisateur.
5) Une fois que la sélection est opérée, un *U-U Download* peut être utilisé par le client pour acquérir le code correspondant au choix du menu.
6) Quand une vidéo est sélectionnée, des messages *U-N AddRessources* sont utilisés pour établir de nouvelles connexions pour les flux audio et vidéo.
7) Les fonctions U-N de manipulation de flux sont alors disponibles à l'utilisateur.

IV.1.4.5.2.3 - Autres interfaces

Il existe aussi d'autre interfaces : l'*interface de base et d'accès* (qui définit les opérations et les attributs qui sont utilisés par les autres interfaces), l'*interface Directory* (qui implémente un service de noms afin d'accéder et de trouver des objets DSM-CC *User-to-user* par leur nom et fournit des facilités pour la navigation), l'*interface de fichier* (qui fournit des opérations simples sur des fichiers, telles que la lecture ou l'écriture de fichiers stockés sur un serveur et qui permet par exemple au client de sauver un profil utilisateur personnel sur le serveur), et les *interfaces d'extension* (qui permettent par exemple à une application d'utiliser des requêtes SQL pour interroger une base de donnée ou qui permettent d'implémenter des mécanismes de sécurité).

IV.1.4.6 - Conclusion

DSM-CC est constitué d'une collection de protocoles permettant d'offrir des services multimédias à travers des réseaux à large bande : un protocole pour établir les sessions et gérer les ressources, des protocoles à la fois pour la configuration du client et le chargement, et des protocoles pour les services applicatifs à large bande grâce à la définition de carrousel de données. DSM-CC fournit également une interface pour le contrôle des flux vidéo de type VCR, et des interfaces pour une multitude d'autres services applicatifs.

Avec ses capacités de configuration et de chargement, DSM-CC est idéal pour les appareils clients à bas coût tels que les *sets top box*. En utilisant DSM-CC, un client est toujours configuré avec les dernières informations sur le réseau, et peut charger à partir du serveur les chargements des dernières versions des logiciels. Dès lors, cela permet d'éliminer les coûts associés au contrôle des versions ou aux versions incompatibles.

IV.1.5 - MPEG-2 Partie 7 (ISO/IEC 13818-7)

Cette partie a été approuvée en avril 1997. Elle spécifie un algorithme de codage audio multi-canaux qui n'est pas contraint d'être compatible en arrière avec MPEG-1 Audio.

MPEG-2 AAC (Advanced Audio Coding) ou NBC (Non-Backward Compatible) est le prolongement des méthodes de codage audio MPEG Layer-3.

Ainsi, on en vante les qualités en lui attribuant la capacité de reproduire avec un fichier compressé en MPEG2 AAC à 96Kbps un équivalent en qualité sur le même fichier qu'un MP3 encodé à 128Kbps.

IV.1.6 - MPEG-2 Partie 8 (ISO/IEC 13818-8)

Cette partie était à l'origine censée être le codage vidéo avec des échantillons d'entrée sur 10 bits. Le travail a été annulé car les industries ne manifestaient pas assez d'intérêt pour une telle norme.

IV.1.7 - MPEG-2 Partie 10 (ISO/IEC 13818-10)

Les normes vidéo, audio et système définissent la structure du signal et le processus de décodage. Un équipement de réception doit être capable de décoder les signaux correspondant à cette définition et restituer un signal correspondant au signal théorique décodé en respectant le processus défini par la norme. Cette définition restant assez vague, le comité de normalisation a inclus dans la norme une partie nommée conformité. Cette norme définit le processus qui permet de vérifier qu'un équipement est bien conforme à la norme MPEG-2, et en particulier les caractéristiques des flux de bits décodables par un équipement conforme à la norme.

IV.2 - APPLICATIONS PRINCIPALES DE MPEG-2

IV.2.1 - DVD

IV.2.1.1 - Généralités

DVD signifie Digital Versatile Disc. Le premier lecteur DVD est apparu au Japon en novembre 1996 et aux Etats-Unis en mars 1997. Le prix de départ des premiers lecteurs était supérieur à 1000 euros, tandis que maintenant on peut en trouver à 100 euros. Le DVD est largement supporté par toutes les grandes compagnies d'électronique, par toutes les grandes compagnies du domaine de l'hardware, et par tous les plus grands studios de films et de musiques. Avec ce support sans précédent, le DVD est devenu le produit électronique de communication qui a rencontré le plus de succès, et ce moins de trois ans après son introduction.
Le DVD permet un rembobinage et avance rapide instantanés, la recherche instantanée des titres, chapitres et des pistes audio. Il est de taille compacte, il est résistant à la chaleur, et il est insensible aux champs magnétiques
Un DVD peut contenir des données audio, vidéo et des données à destination des ordinateurs.

IV.2.1.2 - Vidéo

A des taux supérieurs à 6 Mbps, il n'y a pas de différences perceptibles par rapport au média « maître » encodé sans pertes.

IV.2.1.3 - Audio

On peut multiplexer jusqu'à 8 flux audio avec le même flux vidéo. Chaque flux peut par exemple être dédié à un langage particulier. On utilise le Dolby AC-3 pour les systèmes NTSC et PAL, *MPEG Audio* étant rarement utilisé et uniquement pour les systèmes PAL.

IV.2.1.4 - Spécifications

Video	**ITU-T H.262/ISO-IEC 13818-2 (MPEG-2 Video)** ISO/IEC 11172-2 (MPEG-1 Video)
Audio	**ISO/IEC 13818-3 (MPEG-2 Audio)** ISO/IEC 11172-3 (MPEG-1 Audio) Dolby AC-3 standard
System	**ITU-T H.222 / ISO/IEC 13818-1 (MPEG-2 Systems)** **Program/PES stream only (no Transport streams)**

Représentation codée	MPEG-1 (SIF combo) **MPEG-2 (Main Profile @ Main Level)**
Frame rate	29.97 ou 25 Hz
TV system	525/60 ou 625/50
Aspect ratio	4:3 (all video formats) 16:9 (all formats except 352 pixels/line)
Tailles des frame codées	525/60: 720x480, 704x480, 352x480, 352x240 625/50: 720x576, 704x576, 352x576, 352x288 (MPEG-1 is allowed only in 352x240 or 352x288 res).
Taille de GOP	max 36 fields or 18 frames (NTSC) max 30 fields or 15 frames (PAL)
Taille du buffer	**1.8535008 Mbits (MPEG-2)** max 327689 bits (MPEG-1)
Maximum bitrate	9.8 Mbit/sec

	Single layer Single sided (SL/SS)	Dual layer Single sided (DL/SS)	Single layer Double sided (SL/DS)	Dual Layer Double sided (DL/DS)
12 cm diameter disc	4.7 GB	8.5 GB	9.4 GB	17 GB
8 cm diameter disc	1.4 GB	2.6 GB	2.9 GB	5.3 GB

IV.2.2 - DVB (Digital Video Broadcast)

IV.2.2.1 - Généralités

Le DVB (*Digital Video Broadcast*) est un consortium d'industriels composé de plus de 300 entreprises dans les domaines de la diffusion, de la fabrication, des opérateurs de réseaux, des développeurs de logiciels et des organes de régularisation et ce dans plus de 35 pays. Il a été créé pour définir les standards de transmission pour la télévision numérique. Il est cependant beaucoup plus qu'un simple remplaçant des techniques de transmission de la télévision

analogique existante. DVB fournit une qualité d'image supérieure et la possibilité de voir les images dans un format standard ou dans un format 16 :9, avec du son mono, stéréo, ou surround. Il permet également de nouveaux services comme le sous-titrage, les pistes audio multiples, l'interactivité avec le contenu, les contenus multimédia (on peut par exemple lier des programmes à des matériaux disponibles sur le Web).

Le DVB a repris la norme MPEG-2 pour l'audio, la vidéo et le transport et y a ajouté des guides de programme, des spécifications pour l'accès conditionnel, un canal de retour optionnel pour des services interactifs, et la possibilité d'avoir différents types de paquets.

Le DVB a ainsi défini des tables permettant d'ajouter des informations concernant les programmes multimédias (comme leur nom, les heures de passage) ainsi que des informations concernant les paramètres de réception de ces programmes. Il a également défini un format permettant de transmettre des paquets IP ou ethernet. Toutes ces tables sont emboîtées dans la couche système de MPEG.

IV.2.2.2 - DVB Satellite (DVB-S)

Un canal standard de 8 Mhz peut transporter 5 canaux TV ou 4 canaux de meilleure qualité sans accès conditionnel ou 3 canaux haute qualité avec accès conditionnel.

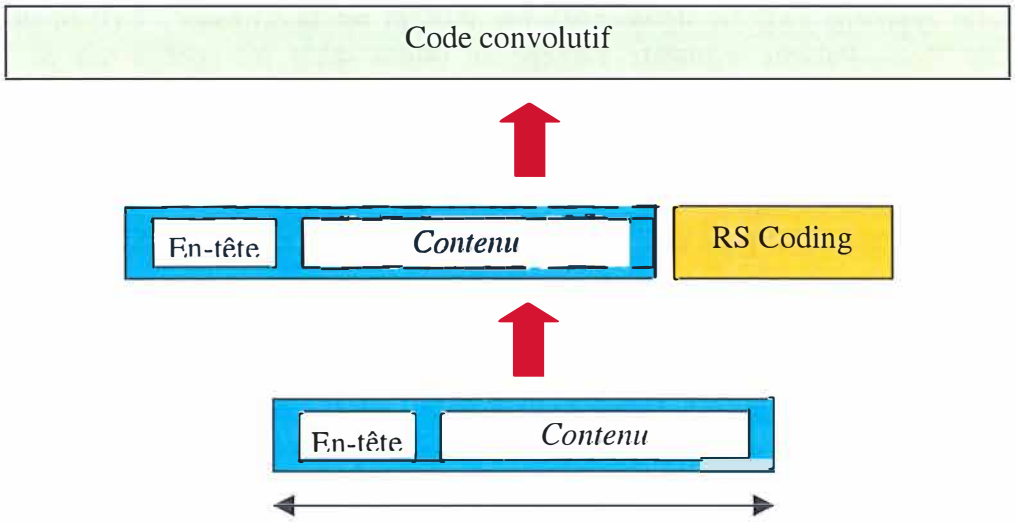
Sample per-multiplex overheads

Stream	bit rate (kbps)
SI	300
PSI	546
Digital Teletext	754
Total per Mux	1600

Bit rate par programme

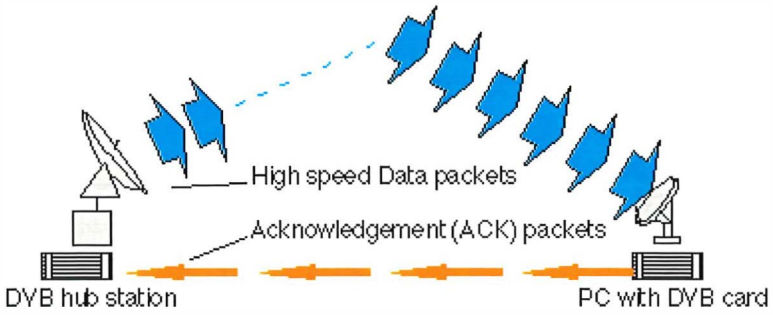
Stream	bit rate (kbps)
TV Video *	5000
Stereo Audio	270
SubTitles	50
Conditional Access	600
Total Programme	5920

La transmission DVB via satellite définit une série d'options pour l'envoi des paquets MPEG-TS sur un lien satellite. Le DVB-S requiert que le paquet de transport de 188 Bytes soit protégé par un code de 16 bytes, le *Reed Salomon Coding*. On applique alors un codage convolutif sur le flux de bits résultant. L'idée du code convolutif est de lier un bit à un ou plusieurs bits précédents de sorte à pouvoir retrouver sa valeur en cas de problème.



Paquet de transport MPEG

La particularité du système DVB-S est qu'il permet d'envoyer les paquets par un système de diffusion à satellite (à grande vitesse) tandis que la confirmation au niveau du client qu'il a bien reçu ces paquets se fait via un canal de retour terrestre.



IV.3 - AUTRES TRAVAUX DANS LE DOMAINE DE LA COMPRESSION

IV.3.1 - PULSENT

Pulsent est capable de délivrer de la vidéo en qualité broadcast à travers une ligne standard de 1,5 Mbits/s (lignes DSL) et est capable de délivrer de la vidéo en qualité VHS à un débit de 384 Kbits/s ou moins. Pulsent peut démontrer qu'il obtient la même qualité d'image à 1,1 Mbps que MPEG-2 à 4-5 Mbps. Ils ont testé leur technologie sur de nombreux matériaux vidéo.

Selon eux, les recherches actuelles basées sur les blocs suggèrent que ce paradigme de la compression a atteint son point maximum. MPEG et les techniques de compression basées sur le bloc arrivent à leur fin.

La nouvelle approche radicale développée par Pulsent est la suivante : au lieu de diviser l'image en blocs, Pulsent segmente l'image en objets selon les spécificités de l'image. Ensuite, on modélise leur mouvement en utilisant des techniques de prédiction. La modélisation efficace de patterns de mouvements communs tels que le changement de taille, la rotation d'objets, les occlusions, les changements de luminosité permet une modélisation effective de la frame suivante à des coûts en terme de bits très faibles. Il en résulte que les objets modélisés matchent plus souvent avec les objets de la frame précédente que lorsque qu'on utilise un matching de blocs.

Bien que l'approche utilisée par Pulsent soit assez différente de MPEG, les *payloads* sont conçus pour être transmis par les protocoles de flux MPEG. On a donc une compatibilité MPEG-2.

Malgré tout, on peut relever quelques inconvénients :

- Est-ce que le marché est prêt à abandonner les investissements en infrastructures actuels ? MPEG-2 est bien ancré parmi les possesseurs de contenus, les opérateurs de services et les fabricants d'électronique.
- Toutes les images ne peuvent pas être segmentées en objets.
  C'est possible de faire du codage vidéo basé sur les objets pour une classe particulière d'images, mais ce n'est pas possible pour des matériaux génériques
- Il y a encore beaucoup à faire dans le domaine du codage par blocs.
- Le fonctionnement en hardware nécessite l'achat d'un chipset.
- Le transcodage de MPEG-2 vers Pulsent n'est pas l'idéal.
- Ils n'ont pas encore trouvé de partenaire industriel.
- Prix de la licence pour la technologie propriétaire.

Computation importante.

Partie 5

Le MPEG-4

V - MPEG-4 (ISO/IEC-14496)

Cette partie consacrée au MPEG-4 a pour source d'inspiration principale, le livre 'Video Compression Demystified' dont vous pouvez trouver la référence dans la bibliographie.

V.1 - INTRODUCTION DE LA NORME MPEG-4

MPEG-4 existe en deux parties, la Version 1 et la Version 2 qui est une amélioration de la Version 1. En d'autres mots, toutes les constructions comprises dans la Version 1 sont également valides dans la Version 2. Le travail commença en 1993, et fut formalisé en juillet 1995 par un appel à proposition. Les différentes propositions pour l'audio et la vidéo furent évaluées par des tests subjectifs et par des experts. Le brouillon 'Draft' de la Version 1 mena à un 'Commitee Draft' en novembre 1997, et devint une norme internationale ISO/IEC en 1999. Pendant ce temps le travail continua sur les extensions de la Version 2 et le 'Draft Amendment' fut gelé en décembre 1999 avec différentes phases d'adoption formelle pendant l'année 2000. Il reste encore pas mal de recherches et de travail à fournir pour rencontrer les besoins du cinéma digital.

La première orientation de la norme MPEG-4 était de pouvoir encoder l'audio et la vidéo à des taux très bas. En fait la norme était particulièrement optimisée pour trois débits de bits :

- en dessous de 64kbits/s
- entre 64 et 384 kbits/s
- entre 384 kbits/s et 4 Mbits/s

La performance à de faibles débits est restée l'objectif principal de cette norme et comme nous le verrons par la suite, des idées très ingénieuses contribuèrent à cet objectif.
Une grande attention fut également portée à la suppression des erreurs ce qui rend le MPEG-4 très adapté pour des environnements difficiles où les erreurs peuvent être nombreuses. D'autres profils et niveaux utilisent des débits jusqu'à 384 Mbits/s, et pour certaines applications ces débits peuvent monter jusqu'à 1.2 Gbits/s.

Plus important encore, MPEG-4 devint bien plus qu'un simple nouveau système de compression, il évolua vers un concept totalement nouveau de compression multimédia utilisant de puissants outils d'encodage permettant une grande interactivité et un très grand panel d'applications cibles.

L'architecture du MPEG 4 est assez ambitieuse. Pour palier aux faibles débits d'Internet, le groupe MPEG a essayé d'enrichir au maximum les fonctionnalités de sa norme MPEG-4.
MPEG-4 a réussi à trouver son identité en offrant une réponse à des besoins émergeants pour des champs d'applications variés, des services audiovisuels interactifs à la télésurveillance en passant par la vidéo en streaming, la compression d'image fixe, l'overlay vidéo, les textures, les scènes 3D,…

Malgré une orientation première vers les faibles débits, MPEG-4 inclut d'importantes extensions pour les studios hi-tech et pour la télévision haute définition.

V.1.1 - L'approche Objet

La plus grande différence par rapport aux systèmes conventionnels de transmission est le concept des *objets*. Différentes parties de la scène finale peuvent être codées et transmises séparément comme *Objets-Video*, ces *Objets-Video* qui seront rassemblés ou *composés* ensuite par le décodeur ce qui entraîne un nombre important de nouvelles possibilités. Différents types d'objets peuvent être traités par différents types de techniques et chacun peut être encodé grâce à des outils lui étant particulièrement adaptés.
Les différents objets peuvent être générés séparément et dans certains cas, une scène peut être analysée pour un certain type d'objet comme par exemple les objets en avant ou en arrière plan.

Les outils servant à représenter les objets visuels naturels avec MPEG-4 doivent provenir d'une technologie standardisée permettant le stockage, la transmission et la manipulation de toutes les données de manière simple et efficace.

Pour atteindre ce but et éviter d'avoir une multitude d'applications non conventionnées qui effectueraient quelques unes de ces fonctions, MPEG-4 propose des solutions et des algorithmes, regroupant la plupart des fonctionnalités demandées par MPEG-4 comme pour :

- la compression des images et des vidéos
- la compression des textures mapping pour les maillages 2D et 3D
- la compression des maillages 2D implicites
- la compression des champs d'animation géométrique des maillages
- l'accès aléatoire de tous types de VO
- l'extension des fonctionnalités de manipulation des images et des séquences vidéo
- le codage des vidéos et des images basé sur le contenu
- le redimensionnement des objets basé sur le contenu
- le redimensionnement spatial, temporel et qualitatif
- la robustesse et la résistance aux erreurs quel que soit l'environnement

Toutes ces solutions sont fournies dans la partie visuelle de la norme MPEG-4.

Les fonctionnalités précédemment décrites montrent la nécessité d'objets audiovisuels (*AVO*). Le codage basé sur une structure orientée objet est nécessaire pour outrepasser les limites des performances actuelles. Globalement, une AVO peut être associé à :

- *un composant vidéo seulement*
- *un composant audio seulement*
- *les deux*

Ainsi, une scène audiovisuelle doit être comprise comme la composition d'AVOs selon un script décrivant leurs relations spatiales et temporelles.

Les caractéristiques spécifiques des composants audio et vidéo des différents AVOs peuvent être très différentes. Le composant audio peut être aussi bien synthétique que réel, mono, stéréo ou multicanal (*surround,...*). Le composant vidéo peut de même être aussi bien synthétique que réel, 2D ou 3D, mono, stéréo ou en vue multiple.

Les nouvelles fonctionnalités de MPEG-4 nécessitent un environnement de représentation ou une architecture qui utilise une structure de données différentes de MPEG-1 et MPEG-2, parce que des parties significatives de l'information visuelle doivent être accessibles pour l'interaction et la manipulation.

L'accès individuel à chacun de ces objets impose à la scène d'être représentée comme la composition d'objets divers qui seront ensuite rassemblés pour recréer la scène.

Dans la suite de l'exposé, nous utiliserons le terme VOP (***V**ideo **O**bject **P**lan*) qui correspond à un composant vidéo de forme arbitraire. Le bloc de définition VOP a pour tâche de définir les objets de la scène qui sont parlants et intéressants et avec lesquels des interactions et des manipulations indépendantes seront possibles. Ceci signifie que ces objets doivent être représentés de façon à fournir un accès simple et de préférence indépendant des autres objets de la scène. Attention, les VOPs n'ont pas nécessairement les mêmes résolutions spatiales et temporelles.

Pour augmenter les possibilités de manipulation, il semble intéressant de considérer des hiérarchies de VOPs associées à différents degrés d'accessibilité (*un VOP peut être divisé en sous-VOPs*). On peut dès lors s'intéresser à aux moins trois types particuliers de VOPs :

- *Un unique VOP (2D)*
  Dans ce cas la représentation du VOP se ramène au cas bien connu d'une scène unique comme savent les manipuler les standards actuels.

- *2 VOPs ou plus mutuellement disjoints, résultant de la segmentation d'une scène 2D*
  Ce cas correspond au partitionnement de la scène en divers VOPs, en utilisant les séquences vidéos 2D traditionnelles en entrée.

- *2 VOPs ou plus, résultant de la composition d'une scène à partir de différentes sources*
  Ceci suppose que les données sont exploitables au début du processus de production. Cette scène est une composition d'objets séparés.

MPEG-4 est la première norme de représentation vidéo tendant à rendre l'utilisateur actif et non plus passif. Et comme l'être humain n'aime pas interagir avec des entités abstraites mais plutôt avec des éléments représentatifs faisant partie d'une scène, le concept de contenu est crucial pour MPEG-4.

Un exemple intéressant fut présenté par une chaîne de pay-per-view, il s'agissait d'un match de football, où l'objet consacré au ballon était dissocié de la scène d'arrière plan. Ainsi seules les personnes ayant payés pouvaient voir le ballon, les autres se contentant de la progression des joueurs sur le terrain.

Le MPEG-4 peut également intervenir lors de la superposition de sous-titres sur une vidéo, en effet les procédés standards comme le DCT ne se débrouillent pas très bien dans ces cas-là, rendant les écritures floues ou ayant leur contours allant se mélanger avec les couleurs du fond. Avec le MPEG-4 on encode le fond et les sous-titres séparément, réalisant ainsi également une économie de bits, il ne reste plus au décodeur qu'à superposer les deux objets.

De plus, plusieurs sous-titres différents peuvent être envoyés dans le flux original, l'utilisateur choisissant celui qui lui convient le mieux.

Nous venons d'identifier trois caractéristiques clés des flux MPEG-4 à savoir :

- *des objets multiples peuvent être encodés en utilisant différentes techniques et ensuite être composés par le décodeur*

- *les objets peuvent être 'naturels' comme les scènes filmées par une caméra ou 'synthétiques' comme c'est le cas pour du texte.*

- *les instructions dans le flux binaire et/ou les choix de l'utilisateur peuvent démarrer différentes présentations à partir du même flux.*

Un point important est par conséquent l'**intégration**. En fait, MPEG-4 souhaite considérer et intégrer harmonieusement des objets audiovisuels naturels et synthétiques incluant l'audio mono, stéréo et multi-canal (Dolby AC-3, ...) ainsi que la vidéo 2D ou 3D en mono, stéréo et en vue multiple. Cette stratégie d'intégration transversale devrait permettre à MPEG-4 de fournir un environnement standardisé où est mise en oeuvre une approche plus globale de la représentation audiovisuelle.

Les derniers points clés concernant MPEG-4 sont la **flexibilité** et l'**évolutivité**. Ces éléments sont essentiels dans le contexte technologique actuel en permanente évolution, et devraient être fournis par un langage de description syntaxique (SDL).

Ainsi MPEG-4 s'adresse t-il à la convergence des applications et la fusion de trois mondes : l'informatique, les télécoms et la télévision.

V.1.2 - Avantages de l'approche Objet

Le MPEG-4 propose des techniques traditionnelles de codage vidéo et audio mais est meilleure que le MPEG-2 car elle est plus efficace et a une meilleure tolérance aux erreurs.
Mais la vraie puissance du MPEG-4 vient des nouvelles applications rendues possibles par l'architecture présentée précédemment. L'encodage indépendant des objets offre un nombre d'avantages. En effet chaque objet peut-être encodé de la façon lui convenant le mieux en émulant au mieux les processus du système psycho visuel humain.

Egalement des facteurs d'échelle spatiaux et temporels peuvent être appliqués, par exemple si les ressources de calcul ou de débit sont limitées, il peut être avisé d'utiliser un maximum de résolution temporelle pour les objets en avant-plan et de ne rafraîchir les objets en arrière plan qu'à un plus faible débit.

V.1.3 - Désavantages de l'approche Objet

Malgré çà il existe également des désavantages à l'approche objet. En effet le décodeur doit être capable de traiter tous les différents flux binaires qu'il supporte et doit avoir la capacité de composition. En conséquence le matériel destiné à un décodeur MPEG-4 sera plus complexe que celui destiné à un décodeur MPEG-2. Et plus de codes seront nécessaires pour

le software destiné au décodeur MPEG-4. Le décodeur devra être plus flexible pour utiliser au mieux les ressources disponibles surtout si elles se trouvent en quantité limitée.

Nous verrons plus loin que MPEG-4 a prévu comme pour le MPEG-2 différents profils et niveaux pour leurs décodeurs.

V.1.4 - Orientation du MPEG-4

La norme MPEG-4 va fournir un ensemble de technologies satisfaisant le besoin des auteurs, des fournisseurs et au final des utilisateurs.

- *Pour les auteurs* : MPEG-4 permettra la production de séquences réutilisables. Il leur permettra une grande flexibilité, autorisant l'amalgame de la télévision numérique, des animations graphiques, et des pages web. En outre, ils auront la possibilité de protéger leurs œuvres.

- *Pour les fournisseurs d'accès Internet* : MPEG-4 offrira des informations transparentes qu'ils pourront aisément adapter à la demande de l'utilisateur (par exemple: l'adaptation en fonction de la langue de l'utilisateur ou encore de changer de publicité lors d'une rencontre sportive), ainsi que le contrôle des transferts (gestion des pertes de données).

- *Pour les utilisateurs* : MPEG-4 aura de nombreuses possibilités qui pourront être accessibles à partir d'un simple terminal. Le but est de permettre *a l'utilisateur* de pouvoir supprimer des informations qu'il ne désire pas ou bien d'accéder à une surcharge d'information (*ex : changement de langage…*). En outre, l'utilisateur peut modifier les attributs de la scène en changeant la position des objets, les rendant visibles ou invisibles, en changeant la police de caractère, la couleur ou encore le volume sonore d'un acteur de la scène. (*par exemple un acteur peut être isolé dans une scène, il sera possible d'isoler également ses dires et de supprimer toute autre source sonore*.).

Voici un large éventail de toutes les applications concernées par les apports d'une telle standardisation :

- La communication temps réel (vidéophone,…)
- La surveillance
- Le multimédia mobile (mini portable faisant office de téléphone, fax, agenda,… par liaison GSM ou satellite)
- Le stockage et la recherche d'informations basés sur le contenu
- La lecture de vidéo sur Internet/intranet sans avoir à télécharger toute la source.
- La visualisation de scène simultanément à plusieurs endroits (téléconférence,…)
- La transmission (tout types de données : vidéo, audio,…)
- La post production (cinéma et télé)
- Le DVD
- Les applications de l'animation de visage : réunions virtuelles,…
- La hiérarchisation et la gestion des objets audio dans une scène.

V.1.5 - Nécessités du MPEG-4

V.1.5.1 - Interactivité basée sur le contenu

- *Outils d'accès aux données multimédia basées sur le contenu*
  MPEG-4 doit fournir un accès aux données et une organisation efficace basée sur le contenu audiovisuel. Les outils d'accès seraient alors les fonctions d'indexation, liens hypertexte, requêtes, affichage, transfert, chargement et suppression.

- *Manipulation basée sur le contenu et édition du flux de données*
  MPEG-4 doit fournir une syntaxe et des schémas pour la manipulation basée sur le contenu et l'édition de flux de données sans que le décodage ne soit nécessaire. Ceci signifie que l'utilisateur devrait être en mesure de sélectionner un objet spécifique dans une scène et d'en changer éventuellement les caractéristiques.

- *Codage des données hybrides (naturelles et synthétiques)*
  MPEG-4 doit fournir des méthodes efficaces pour combiner des images de synthèse à des scènes réelles. Cette fonctionnalité offre quelque chose de nouveau au monde de l'image : l'intégration harmonieuse d'objets audiovisuels naturels et synthétiques. Ceci représente un premier pas vers l'unification de tous types d'informations audiovisuelles.

- *Accès aléatoire temporel amélioré*
  MPEG-4 doit fournir des méthodes efficaces pour l'accès aléatoire à des parties d'une séquence multimédia délimitée dans le temps et avec une bonne résolution.

V.1.5.2 - Compression

- *Codage amélioré*
  Un des objectifs de MPEG-4 est de fournir une qualité audiovisuelle meilleure que celle qu'offrent d'autres standards émergeant, à taux de transfert équivalent.

- *Codage de multiples flots de données concurrents*
  MPEG-4 doit être en mesure de coder de façon efficace de multiples vues ou bandes sons ainsi qu'une synchronisation suffisante entre les flots de données élémentaires résultants. Pour des applications de vidéo multiple ou stéréoscopique, MPEG-4 doit pouvoir exploiter la redondance d'information dans les différentes vues d'une même scène, offrant par ailleurs des solutions compatibles avec la vidéo traditionnelle. Cette fonctionnalité conduit à des représentations efficaces d'objets naturels 3D à condition qu'un nombre suffisant de vues soit disponible. Les applications comme la réalité virtuelle pourront bénéficier de façon substantielle de cette fonctionnalité.

V.1.5.3 - Accès universel

- *Robustesse dans des environnements peu fiables*
  MPEG-4 doit être particulièrement robuste aux erreurs. Et plus particulièrement pour de faibles taux de transfert avec une liaison fortement perturbée. Il est à noter que MPEG-4 est la seule norme de représentation audiovisuelle où les caractéristiques des canaux sont considérées dans la spécification des méthodes de représentation.

- *Multirésolution basée sur le contenu*
  MPEG-4 doit pouvoir prendre en compte les échelles et ainsi offrir une bonne granularité du contenu et améliorer la résolution spatiale, temporelle, la qualité et la complexité. Ceci implique une hiérarchisation des objets dans la scène. La prise en compte de plusieurs échelles pourrait mener à des représentations intéressantes de la scène, où les objets les plus importants seraient représentés dans des résolutions spatiales et/ou temporelles meilleures.

V.2 - LA VIDÉO DANS MPEG-4

Une scène typique va inclure un arrière-plan, un ou plusieurs objets en avant-plan (*des meubles par exemple*), une ou plusieurs personnes et des éléments graphiques. En MPEG-1 ou MPEG-2, la scène complète est échantillonnée une fois par image produisant un bitmap qui est encodé avec les techniques habituelles (*vues dans les chapitres consacrés au MPEG-1 et MPEG-2).*
Le MPEG-4 peut également travailler de la sorte mais il peut également traiter chaque objet séparément, si l'information lui est présentée correctement.

Prenons l'exemple d'une présentation météo dans un bulletin du temps.
Nous avons une *personne*, le présentateur se trouvant devant un *fond bleu ou vert.*
En studio l'image de la personne devant le fond est traitée pour éliminer la couleur du fond et générer un *signal clé* ou *alpha channel* représentant la forme (*et peut-être la transparence*) de la personne en avant-plan. Cette forme est utilisée pour combiner les deux éléments de la scène. L'endroit dans l'arrière-plan (*background*) où se trouve la personne en avant-plan (*foreground*) est remplacé par l'image de la personne. Le reste de l'arrière-plan reste inchangé.

Dans la terminologie MPEG-4, la personne en avant-plan est appelée VO (***Video Object)*** représenté par deux éléments, l'image vidéo de la personne appelée ***Texture*** et le signal clé ou alpha channel appelé ***Forme*** (*shape*). L'image en arrière-plan n'a pas besoin de forme précise, MPEG-4 la considère comme ayant une forme rectangulaire.

V.2.1 - MPEG-4 – La Hierarchie Vidéo

- *VOP (**V**ideo **O**bjetc **P**lan)*

Le point de départ de la vidéo en MPEG-4 est un concept visuel plutôt qu'un signal, comme par exemple un personnage en avant-plan dans une scène. L'objet doit être échantillonné à intervalle régulier, un tel échantillon est appelé VOP (***Video Object Plan***). Donc chaque objet est représenté par une série de VOPs.

- *GOV (**G**roup **O**f **V**ideo Object Planes)*

Un GOV regroupe une série de VOPs, les GOVs sont similaires aux GOPs dans les normes MPEG précédentes. Les GOVs fournissent des emplacements (*points*) dans le flux binaire où les VOPs sont encodés séparément les uns des autres et donc fournissent des points d'accès aléatoires dans le flux binaire.

- *VOL (Video Object Layer)*

Le VOL permet l'encodage d'une séquence de GOVs ou VOPs et ce, à différents niveaux. Le facteur d'échelle appliqué peut-être spatial ou temporel.

VO (**V**ideo **O**bject)
Le niveau VO inclus dans le flux binaire toutes les informations relatives à un objet vidéo particulier.

VS (Video Session)
Le VS est le plus haut niveau dans une scène MPEG-4 et inclut tous les objets vidéo, naturels ou synthétiques d'une scène.

V.2.2 - Encodage de la Forme (Shape)

Il existe deux types de formes pour les objets vidéo dans MPEG-4, les rectangulaires et les arbitraires. *Les formes rectangulaires* sont triviales et désignent principalement l'extension d'une image. Mais il est à noter qu'il s'agit malgré tout d'une amélioration significative de la flexibilité par rapport aux normes précédentes

Les formes arbitraires dans MPEG-4 sont équivalentes à l'alpha channel. A nouveau, la forme représenté l'extension d'un objet vidéo et à chaque point dans le plan de l'image, elle détermine si l'objet associé est visible ou non. La forme est définie sur une région rectangulaire appelée le *Masque*, mis à une taille correspondant aux dimensions horizontales et verticales de l'objet.
Les tailles horizontales et verticales du masque de forme (*Shape Mask*) sont des multiples de 16 pixels.

Les formes arbitraires peuvent être codées comme informations binaires ou en informations grayscale (*échelle de gris*). Les formes binaires sont les plus simples et indiquent pour chaque point si l'objet est transparent ou opaque. Les bords entre l'objet et l'arrière-plan sont toujours fort tranchés. Cela représente normalement une violation du théorème de Nyquist, et l'aliasing (*effet d'escalier*) est très certainement visible. Les formes grayscale, au contraire, permettent douce transition ou créent du flou entre les objets, apportant ainsi un plus grand réalisme.

Les formes binaires sont utilisées pour des applications simples et représentent une transparence ou une opacité totale. Dans les régions de l'image où les objets vidéo sont déterminés comme transparents, ils ne peuvent être vus sous aucune circonstance. L'arrière-plan (*ou le composé de l'arrière-plan et des objets vidéos de plus bas niveaux*) sera vu (*à moins d'être caché par un objets de plus haut niveau*). Où les objets doivent être opaques, l'arrière-plan et les objets de plus bas niveaux seront visibles et les objets vidéos associés à la forme seront visibles à moins d'être cachés par un objet de plus haut niveau (*quelque chose en avant-plan*).

Les formes binaires sont codées en blocs de 16 x 16 pixels, comme les macroblocs utilisés pour les textures, mais sont appelés BABs (***B**inary **A**lpha **B**locks)*. Il y a trois classes de blocs dans un masque binaire :

- ceux où tous les pixels sont transparents (*pas de partie d'un objet vidéo*)
- ceux où les pixels sont tous opaques (*partie d'un objet vidéo*)
- ceux où certains pixels sont transparents et d'autres opaques

L'encodage des deux premiers types est trivial. Les blocs de troisième type sont ceux représentant la frontière de l'objet vidéo et sont encodés grâce à des techniques dérivées du codage arithmétique.

L'algorithme utilisé est appelé CAE (***C**ontext-based **A**rtihmetic **E**ncoding*), étendu pour utiliser la compensation de mouvement. Comme pour l'algorithme conventionnel de codage arithmétique, le codage est basé sur une estimation probabiliste continue. Dans le basique interCAE, l'estimation probabiliste est calculée à partir de 10 pixels au dessus à gauche du pixel à encoder. Pour l'intraCAE où la prédiction et la compensation de mouvement sont utilisées, le contexte inclus certains pixels de l'image courante et d'autres de l'image de référence

Les formes grayscale sont plutôt représentées comme un signal de luminance et sont généralement quantifiées sur 8 bits avec des valeurs de 0 (*totalement transparentes*) à 255 (*totalement opaques*). La forme est encodée en suivant une philosophie similaire à celle utilisée pour les formes binaires. A nouveau il y a des macroblocs qui sont totalement en dehors des frontières de l'objet et d'autres qui sont totalement à l'intérieur de ces frontières. Les blocs en dehors sont marqués comme 'all zero' (*complètement transparents*) et ceux à l'intérieur sont marqués comme 'all 255' (*complètement opaques*). Les blocs qui ne sont ni complètement transparents ni complètement opaques doivent être encodés d'une manière similaire à celle utilisée pour les textures c'est-à-dire en DCT avec compensation de mouvement.

V.2.3 - Encodage de la Texture

Le codage de texture, est le terme MPEG-4 désignant le codage de l'information des images animées conventionnelles et est construit sur le codage MPEG-2 avec des extensions et des améliorations, comme le fut MPEG-2 à partir de MPEG-1 et du JPEG. En fait, un VOP rectangulaire est l'équivalent le plus proche d'une image vidéo (*video frame*) dans la terminologie pré-MPEG-4. Les mêmes concepts d'intra et inter codage sont d'application et les objets vidéo peuvent être codés avec des I-, P-, et B-VOPs. Tous les profils MPEG-4 à l'exception des profils studio utilisent une représentation YUV – 4:2:0 de objets vidéo textures.

Dans MPEG-4 les objets vidéo ne sont pas tous de la même taille, et le codage des textures n'est nécessaire que pour les régions faisant partie de l'objet. Pour les objets rectangulaires, la taille doit être un multiple de 16 pixels (un macrobloc) dans chaque direction, et tous les macroblocs sont traités.

Pour les objets avec une forme complexe, la frontière est définie par *le signal de forme*. L'extension de l'objet est toujours définie par un tableau rectangulaire de macroblocs, mais le codage de texture n'est appliqué que pour les macroblocs se trouvant complètement ou en partie dans l'objet.
Les I-VOPs sont codés comme les I-frames du MPEG-2, mais de nouvelles techniques ont été introduites pour améliorer l'efficacité.
MPEG-4 utilise un *prédicteur* adaptatif pour les valeurs DC

Comme pour le fait que la corrélation des pixels au sein de l'image profite à la prédiction des coefficients DC, cela aide également à l'encodage de certains coefficients AC. Les régions

aux textures similaires, génèrent de tels tableaux de coefficients AC après la transformée DCT.

La similarité (et par conséquent, le bénéfice au niveau du codage) est plus important pour les coefficients les plus significatifs, c'est-à-dire ceux représentant le plus d'énergie de la texture. Ces coefficients sont normalement les coefficients non nuls dans la première ligne de la première colonne. Ce sont également les coefficients qui seront les moins quantifiés et qui utiliseront le plus de bits lors de la transmission et donc ceux qui offrent le potentiel le plus important pour améliorer le codage.

Dans MPEG-4, les coefficients AC de la première ligne ou de la première colonne sont prédits depuis ceux du bloc juste au-dessus ou immédiatement à gauche ou en haut à gauche. La quantification des coefficients est similaire à la méthode utilisée par MPEG-2, mais les mécanismes pour le parcours (*scanning*) des coefficients et pour l'encodage Variable-length ont été améliorés.

La méthode choisie pour lire les coefficients est déterminée par la prédiction :

Quand il n'y a pas de prédiction DC, la méthode Zigzag comme décrite pour le MPEG-2 est utilisée.

Si le coefficient DC à été prédit à partir du bloc de gauche, un système alternatif de scannage vertical (*alternate-vertical scan*) est utilisé, un système permettant de privilégier la direction verticale.

Enfin si le coefficient DC a été prédit à partir du bloc du dessus, un système privilégiant la direction horizontale est choisi (*alternate-horizontal scan*).

Pour améliorer l'efficacité de l'encodage à longueur variable, deux tables VLC sont fournies, et les niveaux de quantification déterminent quelle table sera utilisée. Les codes VLC sont eux-mêmes réversibles, c'est une partie de la stratégie de résistance aux erreurs de MPEG-4. S'il y a une erreur dans le flux binaire reçu, le décodage peut continuer malgré cette erreur. Les données reçues après l'erreur peuvent être décodées en partant de la fin du bloc et en décodant à l'envers les codes VLC jusqu'à trouver l'erreur.

V.2.4 - Encodage de la frontière (Boundary)

La possibilité d'encoder des objets de forme arbitraire crée un problème intéressant aux frontières de l'objet. Les blocs se trouvant en dehors des frontières de l'objet n'ont pas besoin de codage de texture. Ceux se trouvant à l'intérieur des frontières sont codés normalement comme vu précédemment. Le codage de texture est nécessaire pour les blocs frontières, mais l'objet n'existe que sur une partie du bloc. Si nous choisissons de mettre en noir les pixels en dehors des frontières de l'objet cela provoquera de forts pics de fréquence quand le bloc subit la transformée DCT alors que cette énergie ne représente pas une partie intéressante de l'image. Pour éviter cette perte, les blocs sont d'abord bourrés (*padded*). On donne à tous les pixels qui ne font pas partie de l'objet, une valeur égale à la valeur moyenne de tous les pixels faisant partie de l'image. Ce padding (*bourrage*) est raffiné en parcourant les pixels se trouvant en dehors de l'objet et en effectuant une correction basée sur la valeur moyenne de tous les voisins se trouvant dans l'objet. Il faut noter qu'aucune valeur à l'intérieur de l'objet

n'est modifiée. Les modifications des valeurs en dehors de l'objet n'affectent pas le résultat final car ces pixels ne seront jamais visibles. Le procédé décrit minimise l'énergie des coefficients quand le bloc subit la transformée DCT.

V.2.5 - Encodage des objets vidéos aux formes arbitraires

Maintenant que nous avons examiné le codage de différents éléments, il est important de regarder de plus près tous les cas possibles pour le codage de bloc au sein du masque d'un objet vidéo.

- *Bloc Transparent*

Dans le cas le plus simple, *le bloc est marqué comme transparent* dans la partie forme du VOP. Cela signifie que tous les pixels du bloc se situent en dehors des frontières de l'objet et qu'ainsi aucun codage de texture n'est nécessaire.

- *Bloc Opaque*

Le bloc peut être marqué comme opaque dans la partie forme du VOP. Dans ce cas tous les pixels du bloc se trouvent à l'intérieur des frontières de l'objet. Aucun codage de frontière n'est nécessaire mais on a besoin du codage de texture. La texture peut être, selon les cas, intra codée ou prédite au moyen de la prédiction de mouvement.

- *Bloc partiellement Transparent et Opaque*

Les blocs restants sont en partie dans et en dehors de l'objet. Dans le cas d'une forme binaire cela signifie qu'il y a des pixels transparents et d'autres opaques. Dans le cas d'une forme grayscale, cela signifie que tous les pixels ne sont pas transparents et que tous les pixels ne sont pas opaques. Donc dans ce cas, la texture et la forme doivent être encodées.

Ce codage de la texture et de la forme peut être intra ou prédit. Comme dans les systèmes précédents, la syntaxe doit supporter plusieurs possibilités de codage.
Chaque bloc peut être intracodé dans un P-VOP ou un B-VOP ou encore intercodé (prédit).
Dans le cas le plus simple, la différence du vecteur de mouvement (*MVD – **M**otion **V**ector **D**ifference*) est zéro (c'est-à-dire dans le cas où la prédiction est parfaite), et les points du vecteur de mouvement sont une bonne correspondance pour le bloc, ainsi aucun résidu n'est nécessaire. Ce bloc reçoit donc un 'skip code'. Si la correspondance, n'est pas suffisamment bonne, des résidus seront transmis. Dans les autres cas, le MVD sera différent de zéro et les résidus ne seront pas nécessaires.

V.2.6 - Les Sprites

MPEG-4 a un autre type d'objet particulièrement pratique pour les arrière-plans, les sprites.
Un sprite est un objet vidéo plus gros que ce qui peut être affiché à un moment donné. Il est utilisé pour un objet qui est persistant à travers une scène. Les sprites ne sont pas limités à une utilisation dans les jeux, mais cette application montre leur utilisation la plus évidente. En général une scène d'un jeu consiste en un arrière-plan plus un nombre d'éléments synthétiques (en général) qui bougent en accord du script du jeu et dans les actions du joueur. Pendant l'action, la scène vue sera un travelling (pan) autour de l'arrière-plan. Le joueur voit différentes parties à des moments différents, mais elles sont toutes des parties de la même image statique. MPEG-4 fournit la possibilité de transmettre tout l'arrière-plan comme un sprite, et de générer les scènes en envoyant des informations de découpage (*cropping*) ou de

transformation (*warping*) pour déterminer quelle partie du sprite est visible à un moment donné.

Une fois que le sprite à été transmis, seule l'information de *cropping/warping* pour le sprite et les objets à l'avant-plan doit être transmise. Dans un jeu typique, chaque section du sprite sera certainement utilisée plusieurs fois, donc cette approche représente une réduction substantielle des données nécessaires à transmettre.

Transmettre tout le sprite au début peut être très efficient, mais va requérir une quantité de bande passante supplémentaire ou une période de temps avant que l'action puisse débuter. MPEG-4 possède des techniques permettant d'éviter ce problème. Un sprite peut être transmis en section quand cela est nécessaire. La scène d'ouverture va demander que seule la région du sprite nécessaire soit transmise immédiatement. Si la 'caméra' cadre à gauche, une nouvelle partie des données sera transmise à chaque frame, mais elles seront toutes stockées dans le décodeur comme faisant partie du sprite. Ainsi si le cadrage s'inverse (retourne vers la droite) aucune donnée relative à l'arrière-plan, ne devra être retransmise jusqu'au moment où la caméra atteindra l'extrémité droite de la scène originale. De plus, une certaine quantité des données du sprite peut être transmise avec chaque frame pour continuer à construire l'image dans le décodeur.

Cette méthode nécessite qu'un plein écran de données soit transmis avant que la première scène ne puisse être vue. Si cela n'est pas acceptable, le sprite peut-être progressivement encodé, ainsi seule une version à faible résolution est transmise en premier, des régions supplémentaires et une plus haute résolution seront transmises ensuite.

Les sprites sont encodés comme luminance plus deux composants de couleur comme nous l'avons vu précédemment pour les autres versions de MPEG. Egalement les sprites sont toujours intracodés, vu que l'image est essentiellement statique. Mais l'encodage intra profite des techniques améliorées introduites par le MPEG-4.

V.2.7 - Encodage des Textures Statiques

MPEG-4 inclut la possibilité d'appliquer une texture statique sur des formes variées. Par exemple, un logo ou un message peut être appliqué (mappé) sur un poisson en train de nager. L'opération est effectuée dans le décodeur, la forme en déplacement et la texture statique sont transmises comme objets séparés. Dans ce cas la possibilité de pouvoir mettre à l'échelle la texture statique est très importante. Le mouvement de la forme peut causer des déformations (rétrécissement ou expansion) de la texture bien au-delà de sa taille nominale. Pour maximiser la qualité de cette opération, MPEG-4 offre un outil de compression basé sur la technique des ondelettes destiné aux textures statiques. Le point important est que les ondelettes permettent une mise à l'échelle de bonne qualité, les objets ainsi expansés présentent peu d'artéfacts.

V.2.8 - L'Animation

Comme dit précédemment, une des grandes forces du MPEG-4 est la possibilité de transmettre des objets naturels ainsi que des objets synthétiques qui peuvent être composés dans le décodeur. Une des capacités intéressantes de l'utilisation des objets synthétiques est l'animation faciale. Il s'agit d'un autre exemple du mapping de texture sur une forme en mouvement, mais dans ce cas, la forme est spécifiée par un maillage (*mesh*) ou un modèle 3-

D spécifiés par des *nodes*. La position de chaque *node* est encodée et à nouveau l'inter-coding peut-être utilisé pour améliorer l'efficacité de l'encodage lorsque la forme du visage change.

La norme contient toute la syntaxe nécessaire pour coder la position et le mouvement des *nodes*. Elle inclut également des constructions de plus haut niveau comme les 'visual lips' ou 'lèvres virtuelles' permettant de reproduire les phonèmes de la parole.

V.2.8.1 - Les Objets Synthétiques

Les objets synthétiques englobent une importante partie de l'imagerie par ordinateur. Ces objets sont décrits de façon paramétrique, suivant un modèle que l'on peut diviser en 4 parties:

- la description synthétique du visage et du corps humain
- l'animation des champs du visage et du corps
- le codage dynamique et statique du maillage avec les textures
- le codage des textures suivant les vues

V.2.8.2 - Animation du visage

L'animation d'un visage se fait à partir d'un modèle ayant une expression neutre FDP (***Facial Definition Parameter***) contrôlé par une série de paramètres contenus dans le FAP (***Facial Animation Paramètre***). Pour animer un visage, il suffira donc de télécharger le modèle, et d'envoyer les paramètres contrôlant le mouvement du visage qui se traduiront alors sur le visage neutre à l'aide d'un système prévu à cet effet par MPEG-4 (*FIT - **Face Interpolation Technique***). Ce système possède l'avantage de n'avoir besoin que d'un visage neutre permettant d'une part d'accélérer la formation des mouvements, et d'autre part de pouvoir en créer de nouveau sans avoir de modèle d'expression prédéfini.

La partie de la norme relative à l'animation des visages permet d'envoyer des paramètres de calibration et d'animation des visages synthétiques. Ces modèles ne sont pas standardisés par la norme MPEG-4, seuls les paramètres le sont:

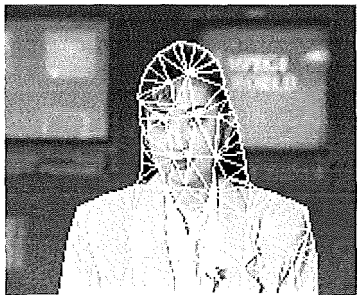
- définition et codage des paramètres d'animation
- positions et orientations des points caractéristiques (points-clefs) pour l'animation du maillage (modélisation 'fil de fer') du visage
- configuration des lèvres correspondant aux phonèmes de la parole
- positions 3D des points caractéristiques
- calibration du maillage 3D pour l'animation
- carte des textures du visage
- caractéristiques personnelles
- codage des textures du visage

V.2.8.3 - Animation du corps

La technologie d'animation du corps proviendra directement de celle du visage, afin de garder l'esprit de standardisation de la norme MPEG-4. C'est une des tâches supportées par la Version 2 de la norme MPEG-4.

V.2.8.4 - Animation des maillages 2D

Le maillage 2D est une partition d'un espace 2D par des polygones eux-mêmes référencés par une liste de nœuds. La norme MPEG-4 utilise uniquement le type de maillage triangulaire, longtemps utilisé pour la représentation d'objets 3D. Ainsi, la modélisation par maillage triangulaire peut être considérée comme la projection d'un maillage 3D sur une image plane, dont voici un exemple:



MPEG-4 a voulu utiliser un maillage dynamique triangulaire pour conserver la facilité de manipulation et les multiples fonctionnalités qu'offre cette solution pour les objets 3D comme:

- *pour la manipulation d'objet vidéo:*
- améliorer le réalisme des scènes
- modifier ou remplacer des objets
- rendre plus robuste l'interpolation spatio-temporelle lors de la reconstruction des images (*en cas de pertes d'information*)

- *pour la compression:*
- le maillage permet d'augmenter le taux de compression avec un faible taux d'erreur

Pour le codage des maillages 2D à structure implicite:

- prédiction basée sur le maillage et transfiguration de texture animée
- modélisation 2D de Delaunay ou maillage régulier avec suivi de mouvement pour les objets animés
- prédiction de mouvement et suspension de transmission des textures avec les maillages dynamiques
- compression géométrique pour les vecteurs de déplacement
- compression de maillage 2D à reconstruction implicite de la structure et du décodeur

Avec une bonne implémentation, cette technique ouvre de fascinantes possibilités. Une personne réelle ou imaginaire peut être représentée par un maillage et une texture. En combinaison avec la synthèse vocale, le visage animé peut être codé pour 'lire' n'importe quel message textuel.

V.2.8.5 - Scalabilité [2]

Le MPEG-4 offre une scalabilité spatiale et temporelle au niveau de l'objet. Dans les deux cas, cette technique est utilisée pour générer un niveau de base représentant la qualité la plus faible supportée par le flux binaire ainsi que un ou plusieurs niveaux supplémentaires de meilleure qualité.

- Ces différents niveaux peuvent être tous produits dans une seule opération d'encodage. La *mise à l'échelle* peut-être implémentée de deux façons distinctes :

- Lorsque l'on sait qu'il y a des limitations au niveau de la bande passante, l'on peut générer un flux binaire ne supportant que le niveau de base et éventuellement un niveau légèrement supérieur.

D'une autre manière, l'on peut envoyer tous les niveaux et le choix se fera au niveau du décodeur. Si la résolution supportée au niveau de l'afficheur est basse ou si les ressources matérielles sont trop faibles, certains niveaux pourront être ignorés.

Il est à noter que la *scalabilité* est appliquée par objet. Cela permet une large gamme de possibilité de codage ou décodage. Par exemple, un décodeur dans un système de jeux peut manquer de ressources de calcul pour décoder tous les objets au taux maximum, il pourra alors choisir de décoder à un faible taux l'arrière-plan mais d'encoder à un taux élevé les objets en avant-plan, permettant une animation la plus fluide possible.

V.2.9 - Les Profils

MPEG-4 fournit un large et riche éventail d'outils pour le codage des objets audiovisuels. Dans le but de permettre une implémentation effective de la norme, des sous ensembles des outils Système, Vidéo et Audio de MPEG-4 ont été identifiés afin de n'être utilisés que pour des applications spécifiques. Ces sous ensembles, appelés "profils", limitent l'ensemble d'outils qu'un codeur aura à implémenter. Pour chacun de ces profils, un ou deux "niveaux" ont été mis en place pour restreindre la complexité de calcul. L'approche est similaire à celle de MPEG-2, où la plus connue des combinaisons Profil/Niveaux est : "Profil principal @ Niveau principal".

Une combinaison "Profil@Niveau" permet :
- à un programmeur de codeur de n'implémenter que les sous ensembles de la norme dont il a besoin, tant qu'il maintient la compatibilité avec d'autres outils MPEG-4 construits sur la même combinaison.
- de tester si ce module MPEG-4 respecte la norme (test de la conformité)

Les profils existent pour différents types de médias (*audio, vidéo et graphiques*) et pour la description de scène. MPEG ne conseille pas de procéder à des combinaisons de ces profils mais toutes les précautions ont été prises pour que les différents types de médias se complètent aisément.

[2] *Scalability* est un terme difficilement traduisible en français, il signifie que la norme est capable de s'adapter facilement dans certains conditions, de se mettre *à l'échelle* pourrait-on dire.

V.2.9.1 - Profils visuels

La partie visuelle de la norme fournit des profils pour le codage des contenus visuels naturels, synthétiques et hybrides naturels/synthétiques. Il y a en tout cinq profils *pour le visuel naturel* :

- *Le profil visuel simple* fournit un codeur, efficace et robuste aux erreurs, d'objets vidéo rectangulaires, adapté pour les applications de réseaux mobiles, tels que PCS et IMT2000.

- *Le profil visuel simple adaptable* ajoute au précédent un support pour coder des objets adaptables au niveau temporel et spatial. Il est très utile pour les applications qui fournissent des services sur plus d'un niveau de qualité à cause du débit ou des possibilités limitées du décodeur, par exemple une application Internet.

- *Le profil visuel 'noyau'* ajoute au profil visuel simple un support pour coder des objets adaptables de forme arbitraire et temporaire. Il est très utile pour les applications telles que celles qui fournissent une interactivité avec le contenu relativement simple (applications multimédias sur Internet).

- *Le profil visuel principal* ajoute au profil précédent un support de codage pour les sprites entrelacés et semi-transparents. Il est utile pour les applications ludiques et interactives de grande qualité comme sur DVD par exemple.

- *Le profil visuel N-Bit* ajoute un support pour coder les objets qui ont des profondeurs pixelliques de 4 à 12 bits. Il est adapté à l'utilisation en vidéo surveillance.

Les profils pour les contenus *visuels synthétiques et hybrides naturels/synthétiques* sont :

- *Le profil visuel d'animation faciale simple* fournit un moyen simple d'animer un modèle de visage, adapté aux applications telle qu'une présentation audio/vidéo pour les malentendants.

- *Le profil visuel adaptable dédié aux textures* fournit des outils pour coder des objets images fixes (texturés) aux dimensions adaptables utilisés pour les applications ayant besoin de multiples niveaux d'adaptation, tels que le plaquage de texture sur un objet dans un jeu ou bien les caméras numériques haute résolution fixes.

- *Le profil visuel basique d'animation 2D* fournit une adaptabilité de l'espace, du SNR et l'animation d'objets fil de fer pour des objets images fixes, ainsi que l'animation simple d'objets visage.

- *Le profil visuel hybride* combine les possibilités du profil visuel 'noyau' vu précédemment et décode également plusieurs objets synthétiques et hybrides, objets image fixe à face simple et animés inclus.

V.2.10 - Résumé des caractéristiques MPEG-4:

- *Représentation de la luminance et de la chrominance au standard Y:U:V* par des pixels régulièrement échantillonnés au format 4:2:0. L'intensité de chaque pixel Y, U ou V est quantifié en 8 bits. La taille et la forme de l'image dépendent de l'application.

- *Codage de multiples VOPs comme image de forme arbitraire* : l'image rectangulaire n'est qu'un cas particulier.

- *Codage de la forme et de la transparence* de chaque VOPs par des séquences d'images planes binaires ou en nuances de gris (*alpha channel*) selon une méthode particulièrement optimisée.

- *Support de I-VOP* (***I**ntra-**VOP***), P-VOP (*temporally **P**redicted-**VOP***) et B-VOP (*temporally **B**idirectionnally predicted-**VOP***) compatible avec les I-, P- et B-Pictures.

- *Support de taux de transfert fixes et variables* des séquences de VOP en entrée.

- *Estimation et compensation du mouvement* propre à chaque VOP, basé sur des blocs 8x8 pixels et des macroblocs de 16*16 pixels.

- *Codage des textures des VOPs* en utilisant une transformation en cosinus discrète (DCT) en bloc de 8x8 ou, alternativement, une DCT adaptée à la forme (Shape Adative DCT) adoptée pour les régions de forme arbitraire, suivie par une quantification et un Run-Length Encoding de type MPEG-1 ou -2.

- *Prédictions efficaces des coefficients de la DCT* pour les I-VOPs.

- *Prédictions du mouvement global* des sprites statiques et dynamiques à partir d'une mémoire panoramique de la VOP utilisant 8 paramètres dédiés.

- *Prise en compte des échelles* temporelles et spatiales des VOPs.

- *Couche de macroblocs adaptable* et marqueurs de mouvement perfectionnés pour la resynchronisation en cas d'erreur.

- *Compatibilité ascendante* avec les algorithmes MPEG-1.

V.2.10.1 - Profils audio

Quatre profils audio ont été définis :

- *Le profil parole* fournit le HVXC qui est un codeur paramétrique de la parole à très faible débit, un codeur CELP bande étroite/bande large et une interface Text-To-Speech.

- *Le profil synthèse* fournit une synthèse par partition utilisant le SAOL et des tables de sons ainsi qu'une interface Text-To-Speech pour produire des sons et de la parole à de très faibles débits.

- *Le profil adaptable* est un super ensemble du profil parole. Il est adapté pour le codage adaptable de la musique et de la parole pour les réseaux tels que Internet et le NADIB (***N**arrow band **A**udio **DI**gital **B**roadcasting*). Le débit est compris entre 6 kBits/s et 24 kBits/s avec des bandes larges entre 3.5 et 9 kHz.

- *Le profil principal* est un super ensemble très riche de tous les autres profils, contenant des outils pour l'audio naturel et synthétique.

V.2.10.2 - Profils graphiques

Les profils graphiques définissent quels éléments graphiques et textuels peuvent être utilisés dans une scène. Ces profils sont définis dans *la partie Système* de la norme :

- *Le profil simple graphique 2D* fournit seulement les outils du BIFS (***B**inary **F**ormat for **S**cene **D**escription*) nécessaires pour placer un ou plusieurs objets dans une scène.

- *Le profil graphique 2D complet* fournit toutes les fonctionnalités graphiques 2D et supporte quelques fonctions comme les graphiques et les textes arbitraires qui peuvent être en conjonction avec des objets visuels.

- *Le profil graphique complet* fournit des éléments graphiques avancés tels que les extrusions et permet de créer une scène avec des lumières sophistiquées. Le profil graphique complet permet des applications telles que des mondes virtuels complexes d'un très haut réalisme.

V.2.10.3 - Les profils de description de scène

Les profils de description de scène, définis dans la partie système de la norme, permettent de créer des scènes audiovisuelles avec seulement de l'audio, du 2D, du 3D ou du 2D/3D mixé. Le profile 3D est appelé VRML car il optimise l'interaction avec le langage VRML :

- *Le profil de scène audio* prévoit un ensemble d'outils du BIFS (***B**inary **F**ormat for **S**cene **D**escription*) pour l'audio seulement. Ce profil supporte des applications de type radio diffusion.

- *Le profil de scène 2D simple* fournit seulement les outils du BIFS pour placer un ou plusieurs éléments audiovisuels dans une scène. Ce profil permet de créer des présentations audiovisuelles mais sans possibilité d'interactions. Il peut être utilisé pour des applications type télé diffusion.

- *Le profil de scène 2D complet* fournit tous les outils du BIFS nécessaires à la réalisation d'une scène 2D. Ce profil est utilisé pour des applications 2D qui nécessitent une interactivité grande et spécifique.

- *Le profil de scène complet* fournit le jeu complet d'outils du BIFS. Ce profil sert à réaliser des applications telles que des mondes 3D virtuels dynamiques et des jeux.

V.2.10.4 - Les profils de description d'objets

Ils comprennent les outils suivants :

- Outil descripteur d'objet (*OD*)
- Outil de synchronisation (*SL*)
- Outil d'information sur les objets (*OCI*)
- Outil de propriété intellectuelle et de protection (*IPMP*)

Actuellement, seul un profil est défini et inclut tous ces outils. La raison principale de la création de ce profil n'est pas de créer des sous ensembles d'outils mais plutôt de leur définir des niveaux. Ceci s'applique spécialement à l'outil de synchronisation des couches MPEG-4 utilisant différentes bases de temps. En introduisant des niveaux, il est alors possible, par exemple, de n'autoriser qu'une seule base de temps.

V.3 - L'ORGANISATION D'UNE SCÈNE MPEG-4

V.3.1 - Description d'une scène

La norme MPEG-4 propose une solution radicalement différente des autres normes pour le codage des vidéos. Les scènes audiovisuelles sont ainsi composées de plusieurs objets médias hiérarchisés.

Ainsi, dans l'arborescence de cette hiérarchie, on trouve:

- des images fixes (background)
- des objets vidéo (objets en mouvement sans background)
- des objets audio (la voix associée à l'objet en mouvement)

MPEG-4 définit donc précisément la manière de décrire une scène. La description d'une scène codée par MPEG-4 peut être comparée au langage VRML dans sa structure et ses fonctionnalités.

Le schéma suivant donne l'exemple d'une scène décomposée suivant l'idée de MPEG-4:

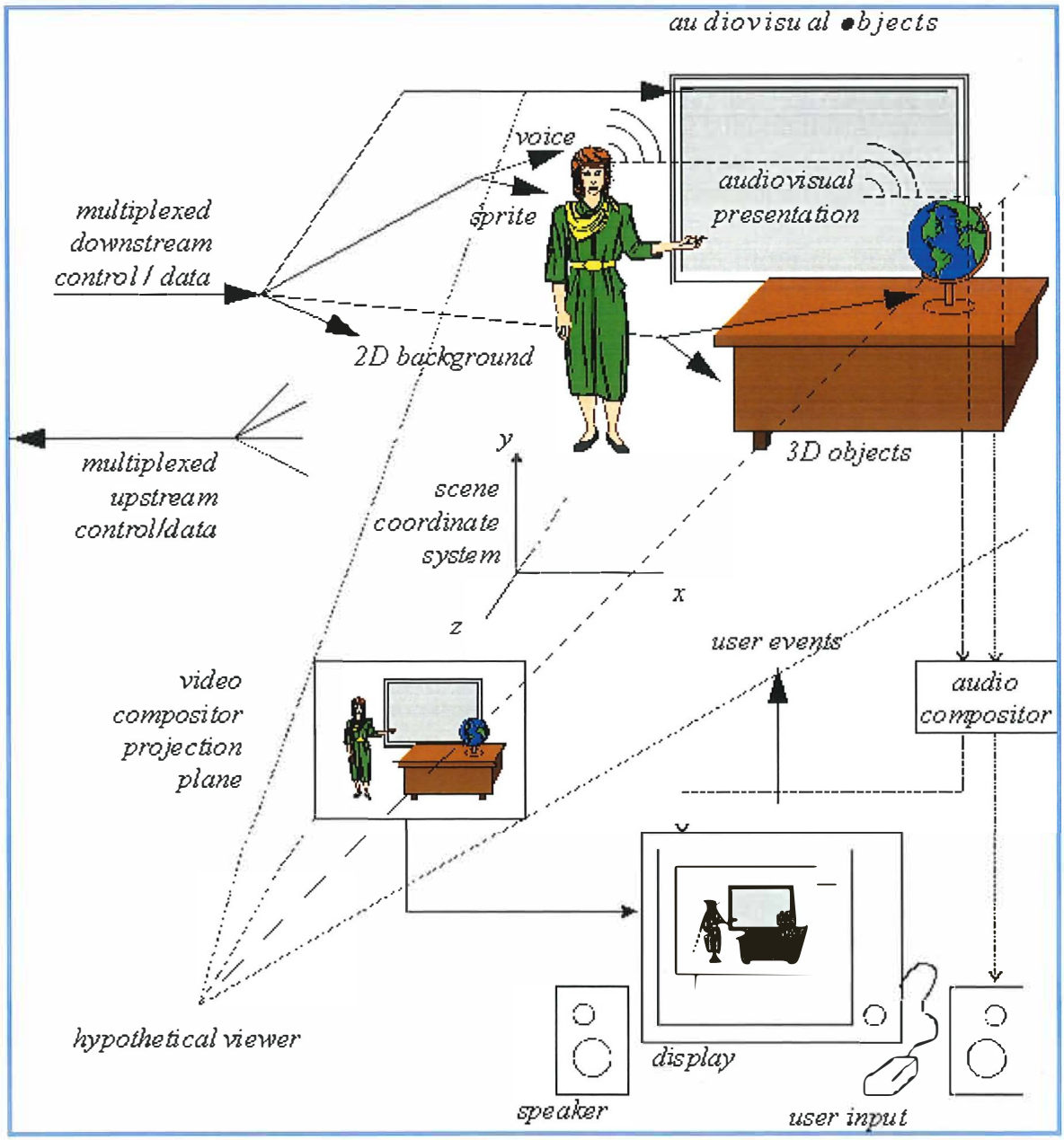


Figure V-1 : Description d'une scène vidéo MPEG-4

Par conséquent, une scène audiovisuelle, codée par MPEG4, est décrite comme un ensemble d'éléments individualisés. Elle contient des composants "média" simples regroupés par type. Ces groupes correspondent aux branches d'un arbre de découpage où chaque feuille représente un élément simple. Voici un exemple de branche de codage, où on distingue les feuilles :

- textes et graphiques
- mouvements de la bouche et son texte associé + animation de la tête
- son synthétique

Par exemple, si cette branche correspondait à une personne qui parle, elle serait divisée en feuilles contenant le fond, la parole et les divers composants graphiques représentant la

personne en train de parler. Une telle construction permet ainsi la construction de scènes complexes tout en autorisant l'utilisateur à ne manipuler qu'une partie des objets. Un objet média peut donc être associé à une information, comme on associe ici la parole à la tête d'un personnage.
Voici le schéma de structure d'une scène MPEG-4 :

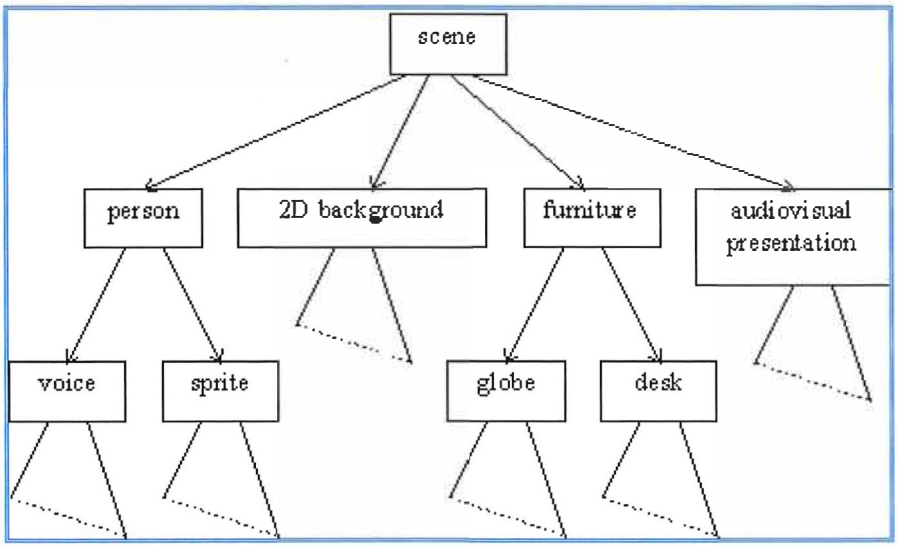


Figure V-2 : Strucutre d'une scène vidéo MPEG-4

MPEG-4 fournit donc des méthodes de codage pour les objets individuels (*comme nous venons de le voir*). La norme permet également d'optimiser le codage de plusieurs objets dans une scène. L'information nécessaire à la composition d'une scène est contenue dans la description de la scène. Celle-ci est codée et transmise avec les objets média. Ainsi, pour faciliter l'interactivité, la description de la scène est codée indépendamment des primitives "Objets média". Une grande attention est portée sur l'identification des paramètres relatifs à la scène. Ces paramètres sont donnés par différents algorithmes qui codent de façon optimale les objets. MPEG-4 autorise la modification de ces paramètres sans avoir à décoder les objets média. Pour cela, ils sont placés dans la partie description de la scène et non avec les objets média.

Plus généralement, MPEG-4 standardise la façon de décrire une scène, en permettant par exemple:

- de placer un objet n'importe où dans un système de coordonnées.
- d'effectuer des transformations géométriques ou acoustiques sur un objet.
- de grouper des éléments "média" simples pour former un composant "média" complexe
- de modifier les attributs d'un objet en transformant ses données.
- de changer, interactivement, la vue et l'écoute d'une scène.

V.3.1.1 - Informations données dans la description d'une scène

- *La première information donne la façon de coder un groupement d'objets.*

Une scène MPEG-4 suit une structure hiérarchique qui peut être représentée comme un graphe acyclique (figure ci-dessous). Chaque feuille du graphe représente un objet média. La structure de l'arbre n'est pas nécessairement statique ; les feuilles (avec leurs paramètres de

positionnement) peuvent être changées. On peut aussi envisager d'en supprimer, d'en remplacer ou même d'en ajouter.

- *La deuxième information donne le positionnement spatial et temporel des objets.*

Dans le modèle MPEG-4, les objets audiovisuels sont à la fois spatiaux et temporels. Chaque objet média a un système de coordonnées locales. Par ce système il est possible d'attribuer un " état " spatio-temporel et une échelle à chaque objet. Les objets média sont disposés dans la scène après avoir subi une transformation du repère local au repère global, transformation définie par un de ses parents.

- *La troisième information donne la valeur qui est attribuée à la sélection.*

Chaque nœud et feuille de l'arbre contient un panel d'informations. Certaines sont accessibles et d'autres restent fixes. Il est donc possible de les paramétrer à loisir suivant les informations données par l'acteur et des contraintes définies par l'auteur.

- *Enfin, la dernière information autorise une autre transformation pour les objets média.*

La structure d'une scène MPEG-4 est fortement influencée par le concept de VRML et ses possibilités d'interaction. Ceci représente l'ambition majeure de MPEG-4.

V.3.1.2 - Interaction avec les objets dans une scène MPEG-4

L'utilisateur visualise en général des scènes respectant le design de leur auteur. Mais suivant la liberté que ce dernier autorise, l'utilisateur a la possibilité d'interagir avec la scène, ce qui lui permet entre autre:

- de changer le point de vue ou d'écoute d'une scène (par la navigation au travers de la scène)
- de modifier la position spatiale d'un objet (*VOP*) dans la scène
- d'appliquer un facteur d'échelle spatial à un objet de la scène
- de changer la vitesse avec laquelle un objet se déplace dans la scène
- d'ajouter des objets
- de supprimer des objets
- de cliquer sur un objet pour obtenir des informations complémentaires sur l'objet ou lui faire effectuer des actions spécifiques.
- de sélectionner une langue parmi celles qui sont proposées
- d'effectuer beaucoup d'autres actions complexes comme établir une communication entre deux personnes par un simple clic de souris….

V.3.1.3 - Le codage des VOPs

Les informations relatives à la forme, au mouvement et à la texture des VOPs sont codées dans des couches VOL (***V**ideo **O**bject **L**ayer*) séparées afin de permettre le décodage séparé des VOPs. Le MPEG-4 VM (***V**ideo **V**erification **M**odel*) utilise un algorithme identique pour coder des informations relatives à la forme, le mouvement et la texture dans chaque couche. Cependant, l'information concernant la forme n'est pas transmise si la séquence qui doit être codée ne contient que des images standard de taille rectangulaire. Dans ce cas, l'algorithme de codage vidéo MPEG-4 a une structure similaire aux algorithmes MPEG-1 et -2. Cela convient à des applications qui requièrent une grande efficacité de codage sans nécessiter des fonctionnalités étendues basées sur le contenu.

L'algorithme de compression MPEG-4 VM est basé sur la technique hybride des DPCM/Transform déjà employée avec succès par les normes MPEG. La première VOP est codée en mode I-VOP. Chacune des images suivantes est codée en utilisant la prédiction inter-image (P-VOP). Seules les données de la plus proche image précédemment codée sont utilisées pour la prédiction. A cela s'ajoute le support des B-VOP. Le procédé de codage est le même que celui des normes MPEG-1 et -2.

En général les images en entrée qui doivent être codées pour chaque couche VOP sont de forme arbitraire, la position et la forme des images varient dans le temps en respect d'une fenêtre de référence. MPEG-4 VM introduit alors le concept de *VOP Image Window* avec une grille de macroblocs adaptable à la forme. Toutes les couches VOL qui doivent être codées pour une séquence vidéo en entrée sont définies en référence à la fenêtre de référence dont la taille est constante. Un exemple de *VOP Image Window* avec sa fenêtre de référence et un exemple d'une grille de macroblocs pour une image VOP particulière sont décrits ci-dessous :

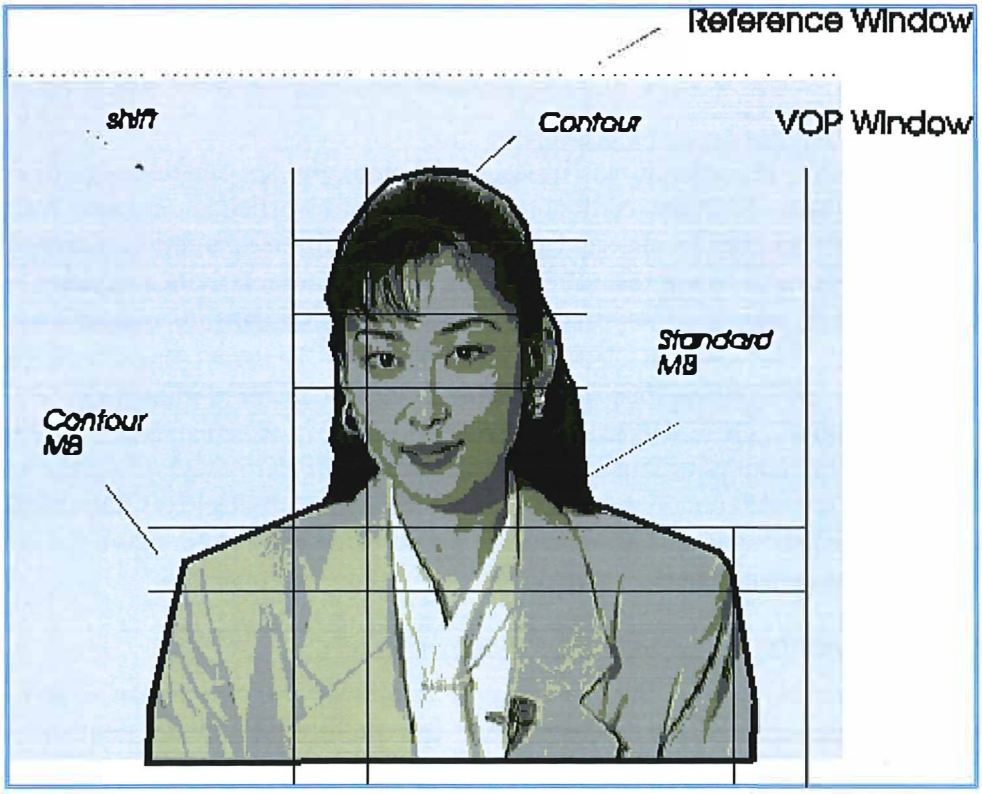


Figure V-3 : Grille de macrobloc MPEG-4 VM

Exemple d'une grille de macroblocs MPEG-4 VM pour une image VOP au premier plan. Cette grille est utilisée pour le codage de l'alpha channel, l'estimation et la compensation de mouvement et le codage de la texture basée sur les blocs et la DCT.

L'information sur la forme d'une VOP est codée avant le codage des vecteurs de position basés sur la grille de macroblocs du VOP et est exploitable aussi bien pour l'encodage que pour le décodage. Dans les étapes suivantes du processus, seules les informations concernant le mouvement et la texture des blocs du macrobloc sont codées (*ce qui inclut les macroblocs standard et les macroblocs de contour indiqués sur la figure ci-dessus*).

- *Codage de la forme*
  MPEG-4 VM repose essentiellement sur deux méthodes de codage concernant la forme. Les techniques qui seront adoptées pour la norme fourniront un codage à

moindre perte des alpha channels et une provision pour les pertes et les informations de transparence, optimisant le compromis entre le taux de transfert et les occurrences de représentation de la forme.

- *Estimation et compensation du mouvement*
  Ces techniques sont utilisées pour explorer les redondances temporelles du contenu vidéo des couches VOPs séparées. De façon générale on peut considérer ces techniques comme une extension de celles utilisées dans la norme MPEG.

- *Codage de la texture*
  Les I-VOPs et les erreurs résiduelles issues de la compensation de mouvement sont codées par une DCT sur des blocs 8x8 de façon similaire à la norme MPEG. La grille de macroblocs adaptable est ici à nouveau employée. Pour chaque macrobloc un maximum de quatre blocs de luminance 8x8 et deux blocs de chrominance 8x8 sont codés. Une adaptation particulière pour les blocs 8x8 représentent les frontières du VOP. Une technique particulière permet de remplir l'arrière-plan hors du VOP avant de lui appliquer la DCT.

- *Codage des textures et des images fixes*
  MPEG-4 utilise les algorithmes basés sur la méthode des ondelettes pour compresser ce type d'images. En effet, cette compression est très efficace quel que soit le taux de transfert, tout en conservant ses capacités d'adaptabilité spatiale et qualitative, ce qui est non négligeable pour résoudre les problèmes d'interactivité (*notamment pour les changement de vue*) et de texturage des objets 2D et 3D dans les images virtuelles.

- *Agencement des informations sur la forme, le déplacement et la texture*
  Tous les "outils" *(DCT, estimation et compensation du mouvement,...)* définis dans les normes MPEG sont pour l'instant supportés par le MPEG-4 VM. L'alpha channel compressé, les vecteurs de mouvement et la DCT sont multiplexés dans une couche VOL du flux en codant les informations sur la forme en premier suivi par le déplacement et, finalement, les textures.

V.3.1.3.1 - Adaptabilité du codage des "objets vidéo"

MPEG-4 code tous les types d'images (*images naturelles rectangles ou objets à contour complexes*). L'adaptabilité de ce codage vient des préférences qu'on peut lui donner en fonction des besoins, comme par exemple:

- réduction de la complexité du décodeur, et donc réduction de la qualité pour des images dont la qualité n'est pas primordiale.
- réduction de la résolution pour une utilisation en petite taille de grands objets
- réduction de la résolution temporelle : séquence vidéo lue avec moins d'images par secondes
- réduction de la qualité sans perte de taille ou de cadence
- ...

Le but du codage MPEG-4 étant de donner à n'importe quel utilisateur les mêmes possibilités d'utilisation, quelles que soient ses capacités matérielles, la norme est donc faite de manière à pouvoir s'adapter aux besoins et aux exigences de l'utilisateur. Ainsi, l'adaptabilité de la norme se fait sur différents champs :

- *Adaptabilité de la complexité au niveau de l'encodeur* pour permettre aux encodeurs de complexité plus ou moins élevée de générer un flux de données valide pour une texture, image ou vidéo donnée.

- *Adaptabilité de la complexité au niveau du décodeur* pour permettre à un flux de données représentant une texture, image ou vidéo d'être décodé par des décodeurs de niveaux de complexités différentes. La qualité de la reconstruction est, en général, relative à la complexité du décodeur utilisé. Ceci pourrait entraîner le fait que des décodeurs moins puissants ne puissent décoder qu'une partie du flux de données.

- *Adaptabilité spatiale* qui permet aux décodeurs de décoder un sous-ensemble du flux de données global généré par l'encodeur pour reconstruire et afficher les textures, images et vidéos à une résolution spatiale plus faible. Pour les textures et images fixes, un maximum de 11 niveaux d'échelonnage spatial est supporté. Pour les séquences vidéo, un maximum de trois niveaux est supporté.

- *Adaptabilité temporelle* pour permettre aux décodeurs de décoder un sous-ensemble du flux de données global généré par l'encodeur pour reconstruire et afficher une séquence vidéo à une résolution temporelle plus faible. Un maximum de trois niveaux sera supporté.

- *Adaptabilité qualitative* qui permet de séparer un flux de données en un certain nombre de couches de façon à ce que la combinaison d'un sous-ensemble de ces couches puisse être décodée en un signal significatif. Cette division au sein du flux de données peut s'effectuer aussi bien au cours de la transmission que dans le décodeur. La qualité de reconstruction est, en général, relative au nombre de couches utilisées pour le décodage et la reconstruction.

Cette adaptabilité permet à tous les utilisateurs du réseau d'avoir accès aux applications temps réel quelle que soit la configuration de leur machine (surtout si celle-ci est limitée).

V.3.1.3.2 - Efficacité du codage

A côté de toutes les nouvelles fonctionnalités et des systèmes de correction d'erreurs, d'une robustesse accrue, le codage de données vidéo avec une grande efficacité du codage à différents taux de transfert continue à être supporté par la norme MPEG-4. MPEG-4 VM autorise le cas particulier d'un seul VOP permettant de coder une simple séquence d'images. Des expérimentations font espérer des améliorations substantielles permettant d'atteindre des taux de transfert inférieurs à 64Kb/s (débit rêvé pour la vidéo-conférence).

V.3.1.4 - Multirésolution temporelle et spatiale

Un but important de la multi-résolution du codage vidéo est d'accroître la flexibilité au niveau du récepteur pour différentes bandes passantes, capacités d'affichage, requêtes sur une banque de données vidéo (*qui permettrait par exemple de parcourir des séquences vidéo*). Une autre capacité de la multi-résolution du codage est de permettre l'existence d'une couche vidéo à transmission prioritaire.

Multi-résolution spatiale

La figure suivante montre la philosophie générale de MPEG-4 pour ce qui est du codage multi-échelle vidéo. Trois couches y sont considérées, correspondant à différentes résolutions

spatiales du VOP. La couche de base (celle dont la résolution est la plus faible) est utilisée pour des taux de transfert médiocres. Les couches supérieures sont alors reconstruites à partir de celle de base et constituent les images de prédiction des séquences en pleine résolution. Si un récepteur n'est pas capable d'afficher un VOP dans sa qualité optimale (pleine résolution), une version plus petite du VOP peut être reconstruite uniquement en décodant le flux de la couche de base.

Multi-résolution temporelle

Cette technique a été développée dans le même esprit que la mutli-résolution spatiale. Un flux de données multi-couches autorise une utilisation à différents taux de transfert. Enfin, la couche supérieure est obtenue par prédiction temporelle à partir des couches plus basses. De cette manière, il est possible de gérer des scènes où les VOP sont affichés à des taux de transfert différents les uns des autres. Par exemple, un personnage au premier plan peut être affiché à un taux de transfert plus important que l'arrière-plan ou d'autres objets de moindre importance

V.3.1.5 - Structure des outils de représentation des vidéos « naturelles » et « synthétiques »

MPEG-4 veut supporter les algorithmes permettant un transfert efficace à très faible taux de transmission (*VLBV - **V**ery **L**ow **B**it-rate **V**ideo, entre 5 et 64kBit/s*) avec un taux de compression satisfaisant, une grande résistance aux erreurs, et une faible complexité pour les applications multimédia temps réel. Toute ces applications prévues pour un faible débit devront être aussi efficaces à haut débit de transfert (*HBV : jusqu'à 4MBit/s*).

Le standard visuel de la norme MPEG-4 permet de coder des images et des vidéos avec des scènes synthétiques créées par ordinateur. A cette fin, le standard visuel contient aussi bien des outils et des algorithmes supportant le codage d'image réelles et de vidéos que des outils supportant la compression de paramètres synthétiques 2D et 3D (*maillages, textes, …*).

V.3.1.5.1 - Fonctionnalités conventionnelles et basées sur le contenu

Le schéma ci-dessous explique la différence entre un codeur VLBV, et un codeur MPEG-4 tenant compte de l'aspect basé sur le contenu:

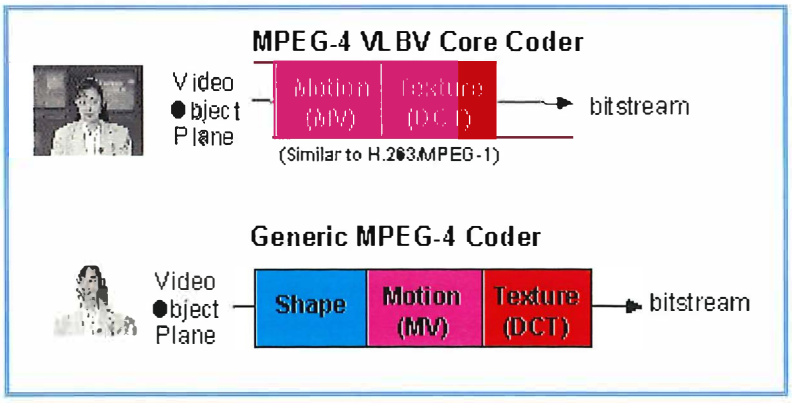


Figure V-4 : Codeur VLBV et Codeur MPEG-4 Générique

Les deux codeurs présentent de nombreuses similitudes, mais le codeur basé sur le contenu possède une extension pour la gestion des formes et de la transparence.

Avantages des fonctionnalités basées sur le contenu:
- codage des images et de la vidéo basé sur le contenu pour permettre un décodage et une reconstruction adaptés à chaque type d'objet vidéo.- accès aléatoire au contenu des séquences vidéos pour permettre des fonctionnalités telles que la pause, l'avance et le retour rapide.- accroissement des possibilités de manipulation du contenu des séquences vidéos pour permettre des fonctionnalités telles que les déformations de textes, textures, images et séquences vidéos synthétiques ou naturelles lors de la reconstruction du contenu de la vidéo.

L'idée d'un codage basé sur le contenu implique que MPEG-4 puisse coder et décoder séparément les différents VOPs d'une scène, afin de permettre une gestion simplifiée de l'interactivité: manipulation et représentation des objets vidéo, ainsi que le mélange entre objets naturels et objets synthétiques (*comme par exemple une scène avec un fond virtuel avec des personnages réels*). Mais les algorithmes supplémentaires nécessaires à la gestion du codage basé sur le contenu ne devront être qu'un ensemble additionnel d'outils aux VLBV et HBV déjà utilisés dans MPEG-1 et 2.

V.3.1.6 - Schéma de codage des images et des vidéos par MPEG-4

Ci dessous le schéma de codage des images et de la vidéo par MPEG-4, qui permet de traiter les images traditionnelles rectangulaires aussi bien que les formes arbitraires d'une séquence vidéo.

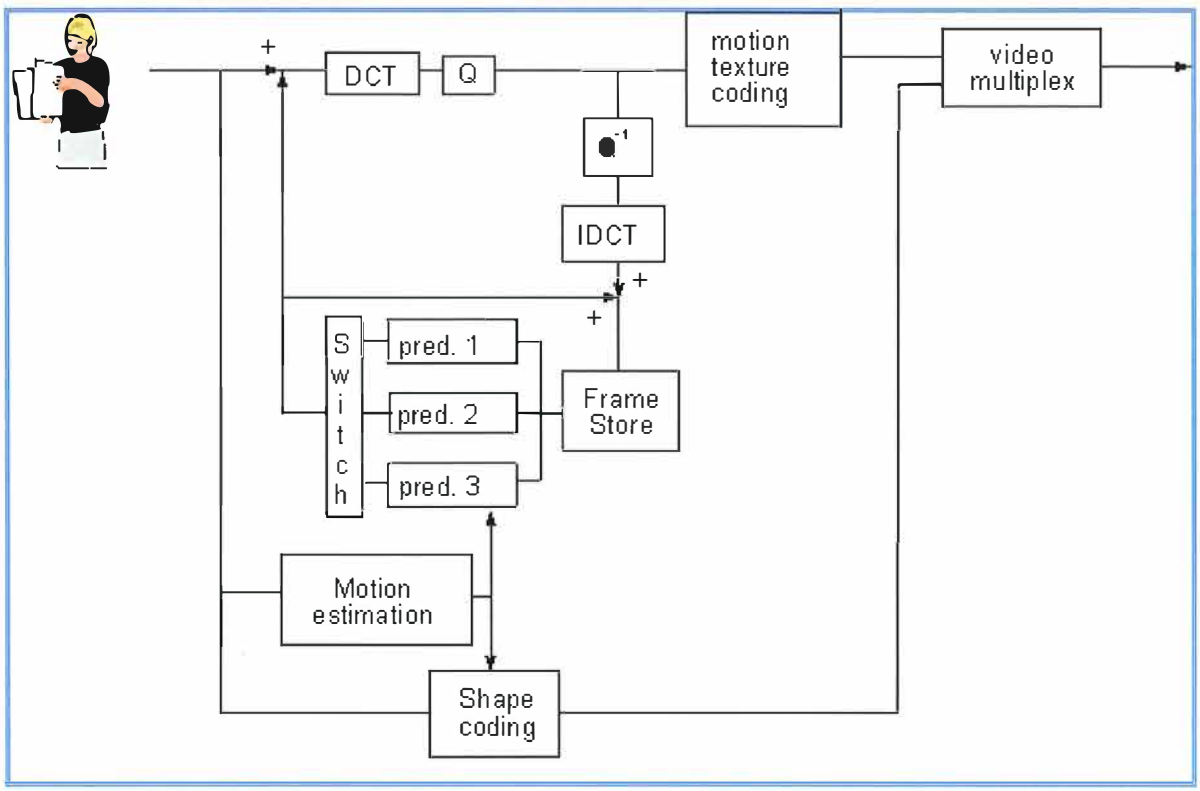
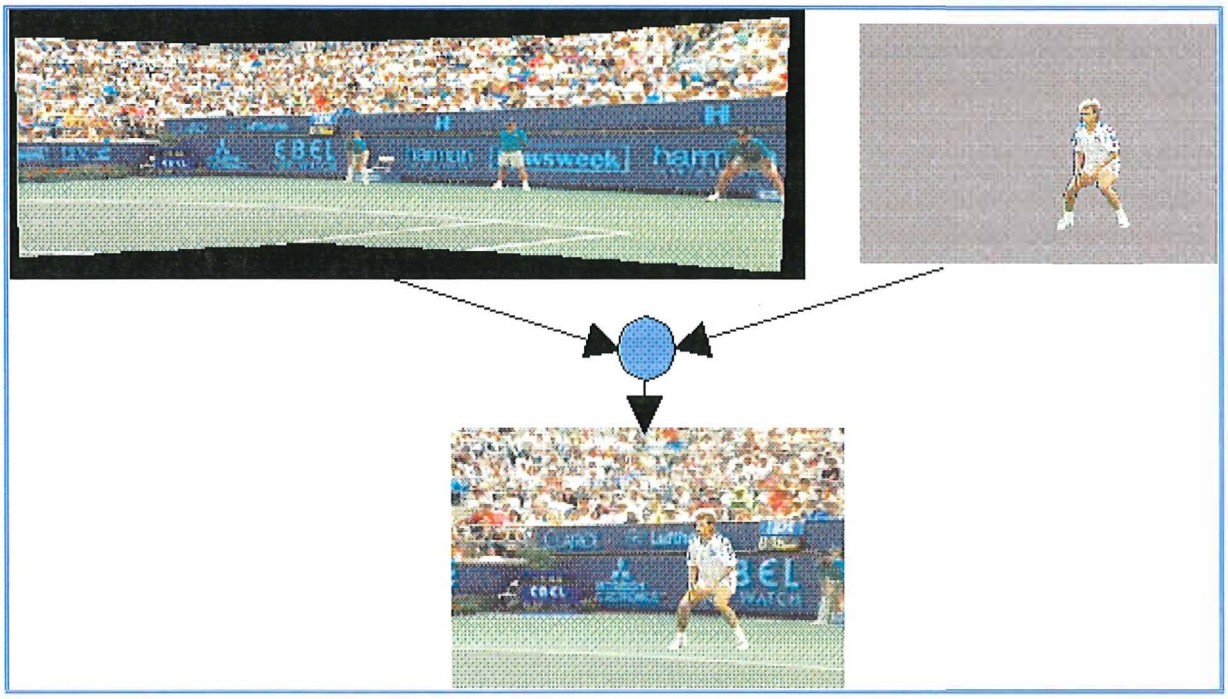


Figure V-5 : Schéma de codage des Images et de la Vidéo par MPEG-4

Le principe du codage MPEG-4 repose sur l'utilisation d'une approche basée sur le contenu. La difficulté étant alors de séparer les objets et le fond d'une scène, pour ensuite en tirer des avantages pour la compression et les fonctionnalités supplémentaires que cela entraînera.

Voyons cela à partir d'un exemple:

Figure V-6 : Exemple d'une scène MPEG-4



Cette image montre bien le concept de base du codage MPEG-4:

- On a isolé le fond de la séquence et recréé un panoramique du fond de la scène complète (estimation et compensation de mouvement par blocs de 8 ou 16 pixels).

- Puis on a extrait du fond le personnage en mouvement

- On ne transfère alors qu'une seule fois le fond, et ensuite le joueur en mouvement.

- Le décodeur recrée ensuite la scène grâce :
  aux paramètres de la caméra pour le fond
  au joueur envoyé dans sa position à chaque image

V.3.1.7 - Echelonnage en fonction des vues

En fonction de la façon dont on regarde une scène, toutes les informations ne sont pas nécessaires. L'échelonnage permet de sélectionner uniquement la partie utile de l'information, et donc de transférer une masse d'informations considérablement réduite entre la base de données et l'utilisateur, données qui seront traitées sous cette forme réduite au codage et au décodage. Cette méthode est de plus applicable aussi bien avec les ondelettes qu'avec le codeur DCT.

V.4 - LES DROITS DE PROPRIÉTÉS INTELLECTUELLES

MPEG-4 traite le problème des droits de propriétés intellectuelles par insertions dans les objets d'un code d'identification (IPI) donnant des informations sur le contenu, le type du contenu et les droits attenant à l'objet en question. Les données contenues dans l'IPI et associées à chaque objet peuvent différer même pour des objets appartenant à une même image (par ex: droits libres sur le fond, mais restreint sur le personnage). L'insertion de l'IPI au moment du codage implique également l'insertion des mécanismes de protection équivalent aux droits sur l'image (protection contre les copies, facturation,...).

PARTIE 6

LE MPEG-7

VI - MPEG-7 (ISO/IEC-15938)

VI.1 - ANALYSE TECHNIQUE MPEG-7

VI.1.1 - Description générale

Le développement de MPEG-7 a débuté en octobre 1996. La norme MPEG-7 est appelée plus formellement *Multimedia Content Description Interface* (interface de description de contenus multimédias). Elle offre des mécanismes permettant d'attacher des métadonnées à des contenus multimédias.

VI.1.1.1 - Qui a développé MPEG-7 ?

MPEG-7 a été développé par des experts représentant des diffuseurs, des fabricants d'électronique, des créateurs et des gérants de contenu, des publieurs et des gérants de droits de propriété intellectuelle, des fournisseurs de service de télécommunication et des académiciens.

VI.1.1.2 - Constat

De nos jours, de plus en plus d'informations audiovisuelles sont disponibles, provenant de nombreuses sources à travers le monde. Aussi, des gens désirent utiliser cette information audiovisuelle pour des raisons diverses. Cependant, avant que cette information puisse être utilisée elle doit être localisée. Cette localisation est d'autant plus difficile qu'il y a de plus en plus de matériaux disponibles et potentiellement intéressants.

L'information audiovisuelle doit être encodée de telle façon qu'elle puisse être utilisable ultérieurement. MPEG-7 offre la possibilité de décrire de manière standardisée les différents types d'informations multimédias. Cette description sera associée au contenu lui-même, pour permettre une recherche rapide et efficace des matériaux pouvant intéresser l'utilisateur.

Le but de la norme MPEG-7 est de permettre l'interopérabilité dans la recherche, le filtrage, l'indexation et l'accès aux contenus audiovisuels en permettant l'interopérabilité entre les appareils et les applications qui se servent des descriptions.

VI.1.1.3 - Frontière de la norme

La norme MPEG-7 ne comprend pas l'extraction (automatique) des descriptions et des caractéristiques. Elle ne spécifie pas non plus le moteur de recherche (ou n'importe quel autre programme) qui peut utiliser cette description.

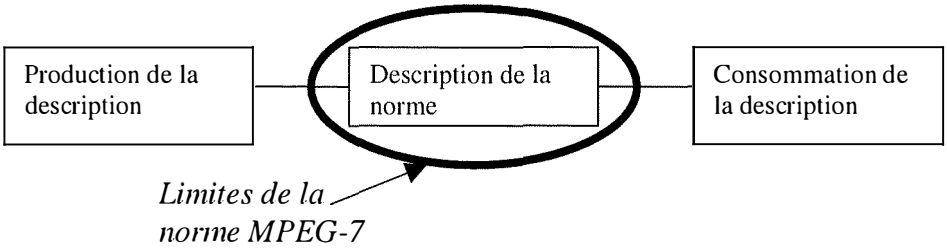


Figure VI-1 : Limites de la norme

Le fait de ne pas prendre en charge l'extraction des données et la conception des moteurs de recherche est dû à la philosophie MPEG. Le but est double. Premièrement cela doit permettre les évolutions technologiques des outils dans ces domaines particuliers. Et deuxièmement cela permet de proposer les meilleurs produits en stimulant le marché et en mettant les concepteurs en concurrence directe.

VI.1.1.4 - Types de données audiovisuelles

Les données audiovisuelles auxquelles on peut attacher des métadonnées MPEG-7 sont des images fixes, des graphiques, des modèles 3D, de l'audio, de la parole, et des informations de composition indiquant la façon dont ces éléments doivent être combinés dans une présentation multimédia (scénarios).

Les outils de description MPEG-7 ne dépendent pas, cependant, de la façon dont le contenu décrit est codé ou stocké. Il est possible de créer une description MPEG-7 d'un film analogique ou d'une image imprimée sur du papier de la même façon qu'un contenu numérisé.

VI.1.1.5 - Lien avec les autres normes MPEG

Bien que les descriptions MPEG-7 ne dépendent pas de la représentation du matériau (codage), MPEG-7 peut exploiter les avantages fournis par le codage MPEG. Si le matériau est encodé selon MPEG-2, on peut insérer une description dans le flux audiovisuel et synchroniser cette description avec le flux. Si le matériau est encodé selon MPEG-4, norme qui fournit les moyens d'encoder un matériau audiovisuel comme des objets ayant certaines relations dans le temps (synchronisation) et dans l'espace (sur l'écran pour de la vidéo, dans une pièce pour de l'audio), il est possible d'attacher des descriptions à des éléments (objets) à l'intérieur de la scène.

VI.1.1.6 - Flexibilité de la description

Les caractéristiques descriptives peuvent être différentes en fonction de l'utilisateur et de l'application. Cela implique que le même matériau peut être décrit de plusieurs façons, selon l'application et l'utilisateur.

Par exemple, un matériau donné peut être décrit par des caractéristiques de bas niveau (la forme, la couleur, la texture pour une vidéo, le tempo pour de l'audio,…), par des caractéristiques de niveaux intermédiaires, ou par des caractéristiques de niveau supérieur donnant des informations sémantiques (« c'est une scène avec un chien qui aboie »).

Le niveau d'abstraction influence la façon dont les caractéristiques pourront être extraites : les caractéristiques de bas niveau pourront être extraites automatiquement tandis que les données de niveaux supérieurs requièrent une intervention humaine.

VI.1.1.7 - Localisation séparée du contenu et de la description

Un contenu et sa description ne doivent pas nécessairement se trouver au même endroit sur le globe. Cette particularité requiert des mécanismes de liens entre un contenu et sa description.

VI.1.1.8 - Nature des informations que l'on peut attacher

Ci-dessous sont listées les informations que l'on peut attacher à un contenu. Toutes ces informations doivent être encodées de telle façon que les opérations sur le contenu soient effectuées de manière efficace.

- Les descriptions à proprement parlé du contenu (caractéristiques de bas niveau, informations structurelles et conceptuelles)
- Des informations sur la forme (le format de codage utilisé, la taille…) : cette information permet de déterminer si le matériau peut être lu par le terminal de l'utilisateur
- Des informations sur les conditions d'accès au matériau : cela inclut le prix ainsi que des liens vers un registre de propriété intellectuelle
- Des informations sur la classification : cela inclut la côte parentale et la classification du contenu dans un certain nombre de catégories prédéfinies
- Des liens vers des matériaux pertinents
- Des informations sur le contexte (essentiellement pour les contenus de non-fiction)
- Des informations décrivant le processus de création et de production du contenu (titre, directeur, équipe de production, …)
- Des informations sur l'usage du contenu (des pointeurs vers des informations de copyright)
- Des informations permettant de naviguer à travers les informations d'une manière efficace (sommaires, variations, …)
- Des informations sur les collections d'objets
- Des informations concernant les interactions du client avec le contenu (les préférences d'utilisation,…)

VI.1.1.9 - Eléments principaux de la norme

Les éléments principaux de la norme MPEG-7 (qui sont détaillés dans la suite de ce document) sont :

- Les Descripteurs (D), qui définissent la syntaxe et la sémantique des représentations des caractéristiques. En d'autres termes, ils lient une caractéristique à un ensemble de valeurs.
- Les Schémas de Description (DS), qui spécifient la structure et la sémantique des relations entre leurs composants. Ces composants peuvent être à la fois des D et des DS. Ce sont les modèles de données de la description. Ils spécifient le type des D qui peuvent être utilisés dans une description donnée, et les relations entre ces D et d'autres DS.
- DDL (langage de définition des descriptions), qui permet la création de nouveaux DS et D, ainsi que l'extension et la modification de DS existants.

- Des outils système, qui s'occupent du multiplexage des descriptions, de la synchronisation des descriptions avec le contenu, des mécanismes de transmission, des représentations codées (à la fois textuelles et binaires) pour un stockage et une transmission efficace, de la gestion et de la protection de la propriété intellectuelle des descriptions MPEG-7, etc.

VI.1.2 - Principales fonctionnalités de MPEG-7

VI.1.2.1 - MPEG-7 Systèmes

VI.1.2.1.1 - Rôles

Cette partie de la spécification inclut les outils nécessaires pour permettre l'encodage et décodage des descriptions, pour préparer les descriptions MPEG-7 pour un transport (*streaming*) et un stockage efficace, et pour permettre la synchronisation entre le contenu et les descriptions. Elle définit également l'architecture du terminal et les interfaces normatives.

VI.1.2.1.2 - Architecture du terminal

L'entité qui fait usage de la représentation codée du contenu multimédia est appelée *terminal MPEG-7*, ou plus simplement *terminal*. Ce *terminal* peut être une application seule ou faire partie d'un système applicatif complet.

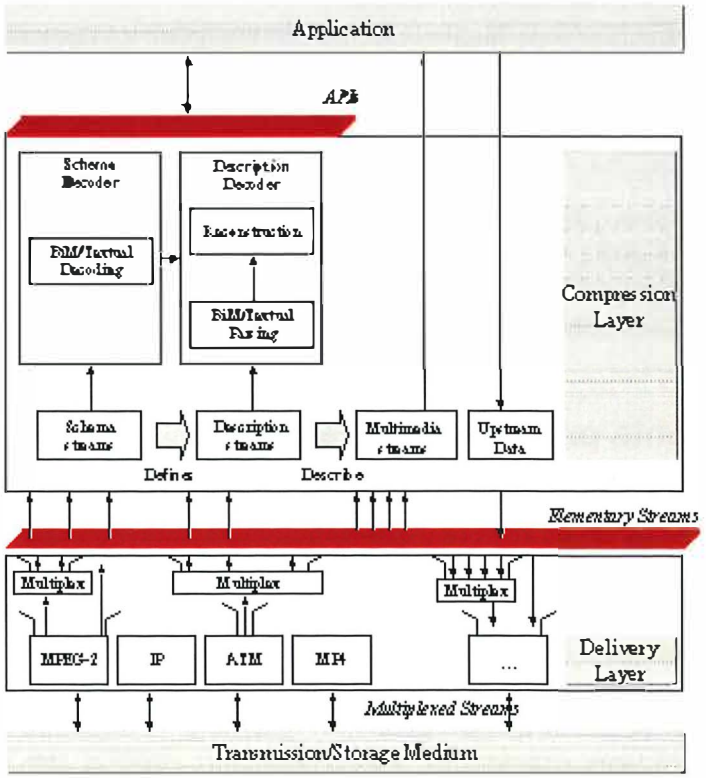


Figure VI-2 : Architecture d'un terminal

La Figure VI-2 montre l'architecture d'un terminal. Au bas se trouve le moyen de transmission et de stockage physique. Il délivre des flux multiplexés à la couche de transport.

Le transport des données MPEG-7 est réalisé par le biais d'une variété de systèmes de transport : flux de transport MPEG-2, IP, fichiers ou flux MPEG-4, etc. La couche de transport offre des mécanismes permettant de synchroniser et de multiplexer des contenus MPEG-7. Tous les flux MPEG-7 ne sont pas descendants (du serveur au client). L'architecture MPEG-7 permet d'acheminer des données (requêtes) en sens inverse, du terminal au serveur.

La couche de transport fournit à la couche de compression les flux élémentaires MPEG-7. Ces flux élémentaires sont des portions de données consécutives individuelles appelées *access units*. Un *access unit* est l'entité de donnée la plus petite à laquelle on peut attacher une information de temps. Les flux élémentaires MPEG-7 contiennent des informations de 2 natures : des informations de schéma (ces information définissent la structure des descriptions MPEG-7) et des informations de description (ces informations sont soit une description complète, soit un fragment de la description d'un contenu multimédia).

VI.1.2.1.3 - Format des données de description

Les données peuvent être représentées soit sous forme textuelle, soit sous forme binaire. Il existe un mappage bidirectionnel entre les deux représentations. L'ordre des éléments est respecté, contrairement à l'ordre des attributs, aux espaces et aux commentaires. La syntaxe du format textuel est décrite dans la partie 2 (DDL) de la spécification tandis que la syntaxe du format binaire est définie dans la partie 1.

VI.1.2.1.4 - Transmission flexible des descriptions

On doit pouvoir transmettre dans un *access unit* un arbre entier ou des fragments d'arbres (chaque fragment étant contenu dans un *access unit)*. On peut également transmettre chaque nœud du graphe dans un *access unit* différent.

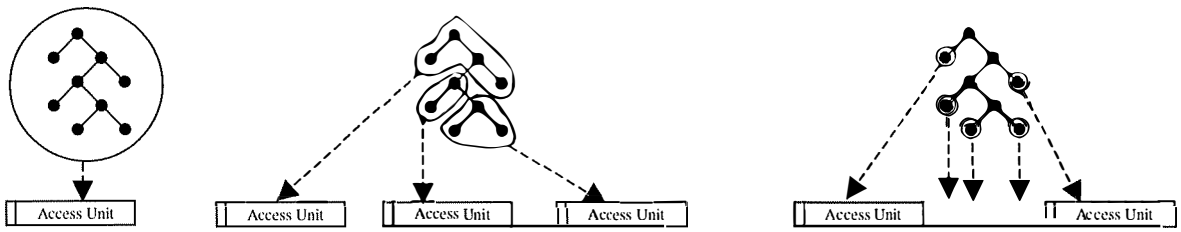


Figure VI-3 : Architecture d'un terminal

On peut avoir besoin d'une granularité plus fine que les *acces units*. C'est le cas notamment lorsque plusieurs sous-arbres ont besoin de la même information de temps. Dans ce cas, un *access unit* peut contenir plusieurs *Fragments Update Unit* (FUU).

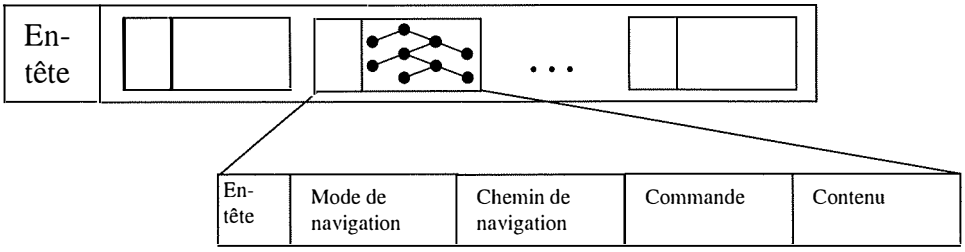


Figure VI-4 : Fragment Update Unit

Chaque FUU est constitué :

- d'un champ *Mode de navigation*, qui spécifie le type d'adressage
- d'un champ *Chemin de navigation*, qui spécifie le chemin vers le nœud destination
- d'un champ *Commande*, qui spécifie le type de mise à jour effectuée (ajout, suppression ou remplacement d'un nœud ou d'un contenu)
- d'un champ *Contenu*, qui contient le fragment de description qui doit être ajouté ou remplacé

La Figure VI-5 montre les différentes possibilités de navigation dans les arbres de description.

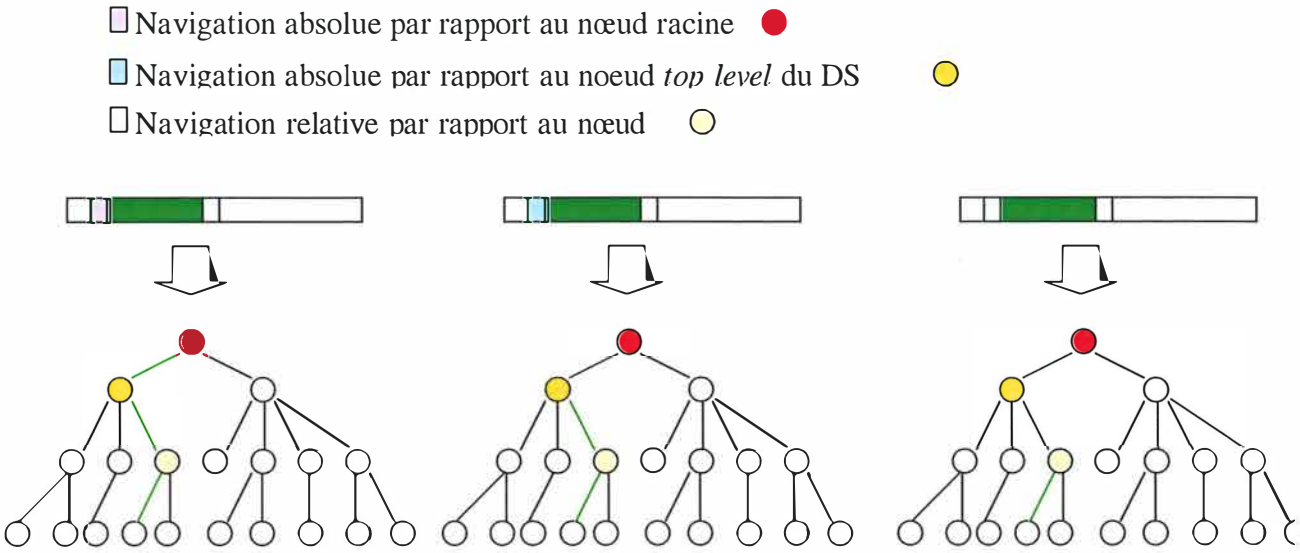


Figure VI-5 : Possibilités de navigation dans les arbres de description

La Figure VI-6 illustre la mise à jour dynamique par des *Fragment Update Commands*.

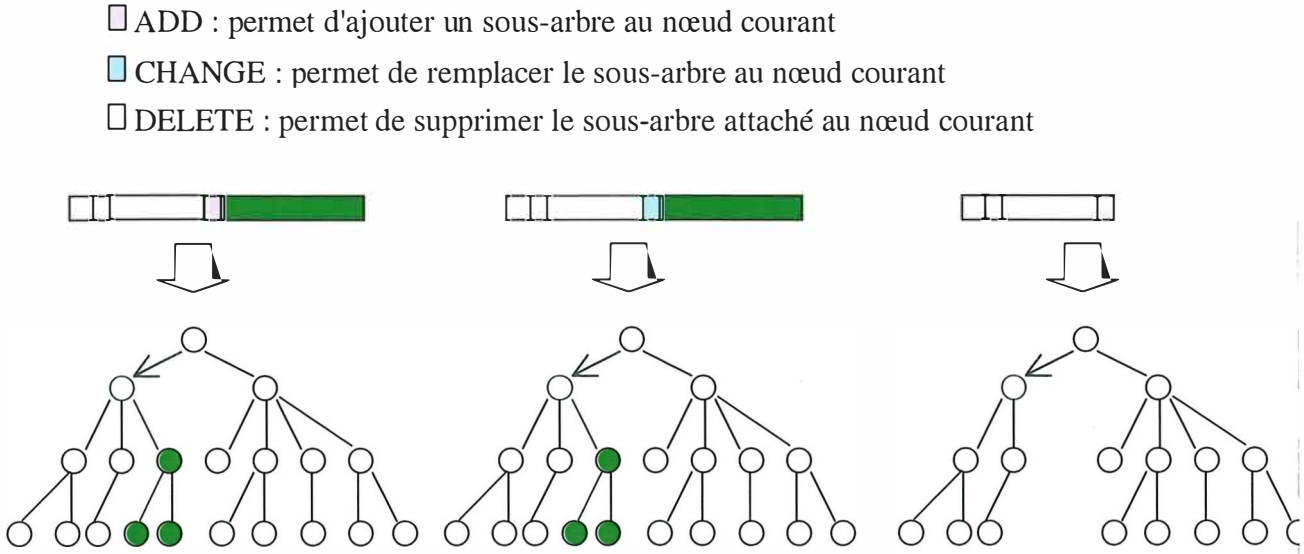


Figure VI-6 : Mise à jour dynamique

VI.1.2.1.5 - BiM (compression des descriptions)

L'introduction de BiM vient du fait que les descriptions sont assez larges en terme de place et qu'elles doivent être transmises dans un environnement où la bande passante est limitée. De plus BiM permet un accès aléatoire rapide. Ce sont deux caractéristiques cruciales étant donné que les descriptions sont volumineuses.

XML (le langage choisi pour les descriptions) n'a pas été conçu pour les environnements en temps réel. Les pages HTML utilisent un nombre limité de tags et l'overhead dû à ces tags (en représentation textuelle) n'est pas un facteur critique quant aux performances de la transmission. Par contre, l'overhead introduit par les structures MPEG en représentation textuelle peut influer de manière critique sur les performances de transmission. C'est pour cela qu'a été crée BiM (Binary format for MPEG-7). Il permet de compresser de manière drastique n'importe quel document XML.

Le format binaire a deux propriétés importantes. Premièrement, les redondances structurelles sont supprimées, par la connaissance des schémas (nom des éléments, nom des attributs, etc.). De ce fait, la structure du document subit une compression de l'ordre de 98% en moyenne. Deuxièmement, les valeurs des attributs et des éléments sont encodées par le même *codec*. L'encodage des types de données spécifiques permet une compression générale de l'ordre de 80%.

Les capacités de transmission de BiM permettent de scinder un document XML volumineux en plusieurs morceaux. Ces morceaux peuvent être délivrés séparément. Le décodeur du client ne doit pas nécessairement télécharger le document XML en entier pour le traiter. Cela permet de réduire à la fois la mémoire nécessaire au niveau du décodeur et de réduire la consommation en bande passante.

VI.1.2.2 - MPEG-7 DDL

Le *langage de définition des descriptions* (DDL) est une partie centrale de la norme MPEG-7. Il permet de créer ses propres *descripteurs* (D) et *schémas de description* (DS). Il permet aussi l'extension et la modification de DS existants. Le DDL définit des règles syntaxiques pour construire et combiner les D et DS.

Le DDL est basé sur XML Schéma. XML Schéma a été créé dans le but d'étendre les fonctionnalités offertes par le mécanisme des DTD associées à des fichiers XML. XML Schéma permet d'appliquer des constructeurs particuliers pour contraindre un document XML: contraintes sur les éléments et leur contenu, les attributs et leurs valeurs, les cardinalités et les types de données.

DDL s'articule comme suit : les composants de structure XML Schéma, les composants de types de données XML Schéma et les extensions spécifiques à MPEG-7.

VI.1.2.2.1 - XML Schéma : Structure

XML Schéma : Structure permet de décrire la structure et les contraintes sur le contenu des documents XML 1.0. Un schéma XML peut contenir des composants primaires ou secondaires.

VI.1.2.2.1.1 - Les composants primaires

- Le *Schéma* est une enveloppe qui contient des définitions et des déclarations.

```
<xs:schema xmlns:xs="http://www.w3.org/1999/XMLSchema"
targetNamespace=http://www.mpeg.org/mpeg-7 version="1.1">
</xs:schema>
```

- Les définitions de type simple, qui définissent des types de données simples, et qui ne peuvent pas avoir des éléments enfants ou des attributs.

```
<simpleType name="US-State" base="string">
    <enumeration value="AK"/>
    <enumeration value="AL"/>
    <enumeration value="AR"/>
    ......
</simpleType>

<attribute name="State2" type="US-State"/>
```

- Les définitions de type complexe, qui peuvent avoir des attributs et des éléments enfants ou être dérivés d'autres types simples ou complexes.

```
<complexType name="personName">
    <element name="title" type="string"/>
    <element name="forename" type="string"/>
    <element name="surname" type="string"/>
    <attribute name="age" type="integer"/>
</complexType>

<element name="producer" type="personName"/>
```

- Les déclarations d'attributs

```
<attribute name="State1" type="string"/>
```

- Les déclarations d'éléments

```
<element name="myglobalelt1" type="mySimpleType"/>
<element name="myglobalelt2" type="myComplexType"/>
<element name="myglobalelt3">
    <complexType>
        <element name="mylocalelt" type="otherType"/>
        <element ref="myglobalelt2"/>
        <attribute name="mylocalattr" type="date"/>
    </complexType>
</element>
```

Les définitions de types définissent des composants de schéma internes qui peuvent être utilisés dans d'autres composants de schémas (tels que des déclarations d'attributs et d'éléments).

VI.1.2.2.1.2 - Les composants secondaires

- Les définitions de groupes d'attributs

```
<attributeGroup name="myAttrGroup">
    <attribute name="myD1" type="string"/>
    <attribute name="myD2" type="integer"/>
    <attribute name="myD3" type="date"/>
</attributeGroup>

<complexType name="myDS">
        <element name="myelement" type="myType"/>
        <attributeGroup ref="myAttrGroup"/>
</complexType>
```

- Les définitions de groupe

```
<group name="myModelGroup">
    <sequence>
        <element name="firstThing" type="type1"/>
        <element name="secondThing" type="type2"/>
    </sequence>
</group>

<complexType name="newType"
    <choice>
        <group ref="myModelGroup"/>
        <element ref="anotherThing"/>
    </choice>
</complexType>
```

VI.1.2.2.2 - XML Schéma : Types de données

XML Schéma : Types de données permet de définir des types de données qui seront utilisés pour contraindre les éléments et les attributs dans les Schémas XML. Il fournit un ensemble de types de données primitifs, un ensemble de types de données dérivés et des mécanismes par lesquels l'utilisateur peut définir ses propres types de données dérivés.

VI.1.2.2.3 - Extensions MPEG-7

Afin de satisfaire aux exigences spécifiques à MPEG-7, des caractéristiques ont été ajoutées, comme les types de données de tableaux et de matrices (de taille fixe ou paramétrable).

VI.1.2.3 - MPEG-7 Audio

VI.1.2.3.1 - Outils de description audio de bas niveau

Cette partie décrit des outils de bas niveau servant de base pour la construction d'applications audio de niveau supérieur. Les D audio de bas niveau ont une grande importance dans la description audio. Il en existe 17. Parmi ces D, il en existe un simple mais utile, le D de silence. Il attache la sémantique de silence (pas de son significatif) à un segment audio. Il peut être utilisé pour faciliter une segmentation future du flux audio, ou permet de suggérer de ne pas traiter le segment.

Il existe 2 façons de décrire des caractéristiques audio de bas niveau : soit en échantillonnant des valeurs à des intervalles réguliers, soit en utilisant des *segments* (voir partie MDS).

VI.1.2.3.2 - Outils de description audio de haut niveau

Quatre ensembles d'outils de description audio sont définis.

VI.1.2.3.2.1 - Les outils de description du timbre des instruments de musique

Leur but est de décrire les caractéristiques perceptuelles des sons des instruments. Le timbre est défini dans la littérature comme les caractéristiques conceptuelles qui font que 2 sons qui ont la même hauteur et le même bruit (*pitch*) semblent différents. Ces D utilisent des notions telles que la clarté ou la richesse d'un son.

VI.1.2.3.2.2 - Les outils de reconnaissance du son

Les D et DS de reconnaissance du son constituent une collection d'outils pour l'indexage et la catégorisation de sons généraux, avec application directe aux effets sonores.

VI.1.2.3.2.3 - Les outils de description de contenu parlé

Les outils de description de contenu parlé permettent une description détaillée de mots parlés dans un flux audio. Bien que reconnaissant que les technologies actuelles de reconnaissance de la parole automatique (ASR) sont limitées et que l'on trouvera toujours des mots en dehors du vocabulaire, les outils de description de contenu parlé sacrifient la compacité à la robustesse de la recherche.

Les outils de description de contenu parlé sont divisés en 2 unités fonctionnelles: le treillis, qui représente le décodage produit par un moteur ASR, et l'en-tête, qui contient l'information sur la personne qui parle qui a été reconnue et sur le reconnaisseur lui-même.

VI.1.2.3.2.4 - Les outils de description de mélodies

Le *DS Melody Contour* est une représentation compacte de l'information mélodique. Il permet un matching de similarités mélodiques efficace et robuste, par exemple lors des *query-by-humming*.

VI.1.2.4 - MPEG-7 Visual

Les outils de description visuelle MPEG-7 sont répartis en deux catégories : les structures de base et les D qui couvrent les caractéristiques visuelles de base (couleur, texture, forme, mouvement, localisation, etc.).

VI.1.2.4.1 - Structures de base

VI.1.2.4.1.1 - Grid Layout

Cet outil permet de diviser l'image en un ensemble de régions rectangulaires de tailles égales, afin que chaque région puisse être décrite séparément. Chaque région (ou sous-ensemble) de la grille peut être décrite par d'autres D (D de couleur et de texture par exemple).

VI.1.2.4.1.2 - Coordonnées spatiales 2D

Cette description définit un système de coordonnées spatiales 2D (ainsi qu'une unité) qui sera utilisé dans d'autres D/DS lorsque cela sera nécessaire. Un des avantages de ce D est qu'on ne

doit pas changer la description MPEG-7 si on modifie la taille de l'image. Dans ce cas, seule la description du mapping entre l'image originale et l'image éditée est nécessaire.
Il existe deux types de systèmes de coordonnées: le système local, dans lequel toutes les images sont mappées par rapport à la même position, et le système intégré, dans lequel chaque image (d'une vidéo par exemple) peut être mappée dans des régions différentes en se référant à la première image. Ce système peut être utilisé pour représenter les coordonnées d'une mosaïque.

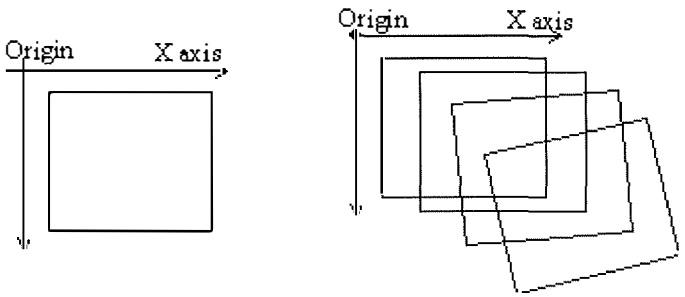


Figure VI-7 : Système de coordonné local et intégré

VI.1.2.4.1.3 - Interpolation temporelle

L'interpolation temporelle peut s'avérer utile pour approximer des valeurs variables multidimensionnelles qui changent dans le temps, comme par exemple la position d'un objet dans une vidéo. La taille de la description de l'interpolation est généralement beaucoup plus petite que la description de toutes les valeurs.

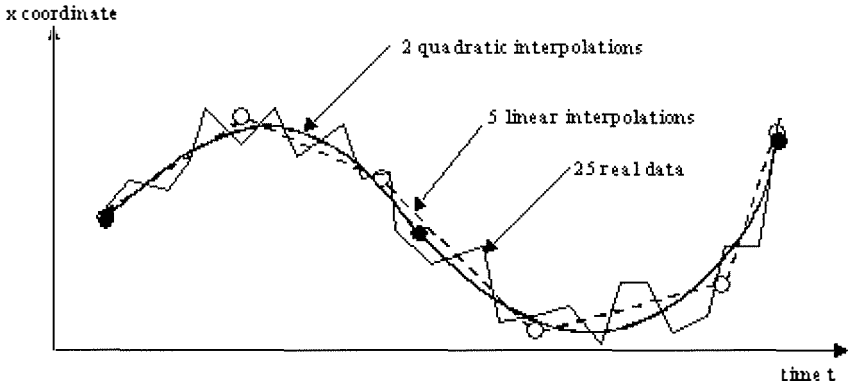


Figure VI-8 : Interpolation temporelle

VI.1.2.4.2 - D de couleur

VI.1.2.4.2.1 - Espace des couleurs

Il permet de caractériser l'espace de couleur qui est utilisé dans d'autres D basés sur la couleur. Les espaces de couleurs supportés sont entre autres RGB et YCrCb.

VI.1.2.4.2.2 - Quantification de la couleur

Ce D définit une quantification uniforme d'un espace de couleurs. Afin d'avoir un sens dans le contexte MPEG-7, ce D doit être combiné avec d'autres descripteurs. Il peut par exemple être combiné avec le descripteur de couleur dominante pour expliquer la signification des valeurs des couleurs dominantes.

VI.1.2.4.2.3 - Couleurs dominantes

Ce descripteur permet de représenter des caractéristiques locales (caractéristiques d'un objet ou d'une région dans une image) lorsque quelques couleurs sont suffisantes pour caractériser les informations de couleur de la région concernée. Ceci est applicable également à des images entières, comme par exemple des images de drapeaux ou des images de marques.

VI.1.2.4.3 - D de forme

VI.1.2.4.3.1 - Forme basée sur la région

La forme d'un objet peut être représentée par une région unique, un ensemble de régions, ou par les trous à l'intérieur d'un objet (comme illustré dans la Figure VI-9). Etant donné que le descripteur utilise tous les pixels constituant la forme dans la frame, il peut décrire n'importe quelle région, c'est-à-dire pas seulement une forme simple de régions connectées (a, b) mais aussi une forme complexe constituée des trous dans un objet ou de plusieurs régions disjointes (c, d, e).

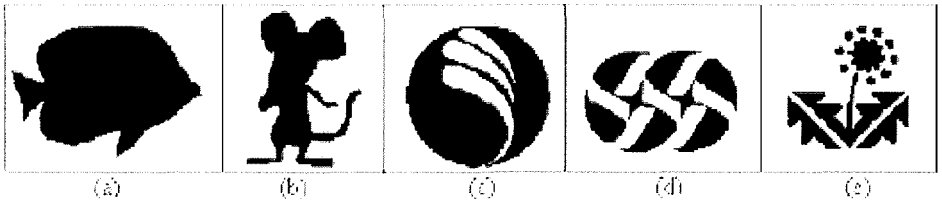


Figure VI-9 : Exemple de régions

Ce descripteur permet de décrire ces formes de façon efficiente (la taille des données pour cette représentation est fixée à 17,5 bytes) dans un descripteur unique, et est robuste par rapport à des déformations mineures aux abords des objets. Ce D est caractérisé par sa petite taille, sa facilité d'extraction et son matching rapide.

Le concept de robustesse est illustré dans la Figure VI-10. (G, h, i) ont des formes très similaires pour une tasse. Les différences se situent au niveau de la poignée. Le descripteur considère g et h similaires mais différents de i parce que la poignée est remplie. De façon similaire, les figures (j, k, l) où l'on voit deux disques séparés, sont considérées comme similaires du point de vue de ce descripteur.

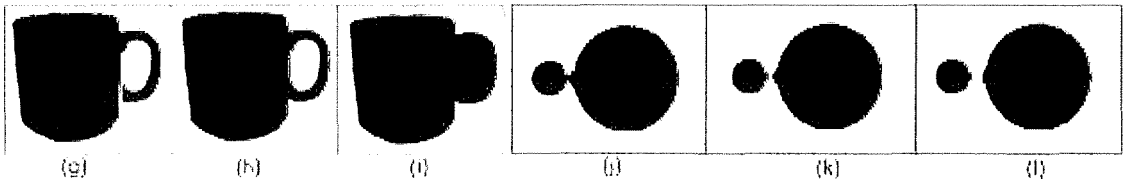


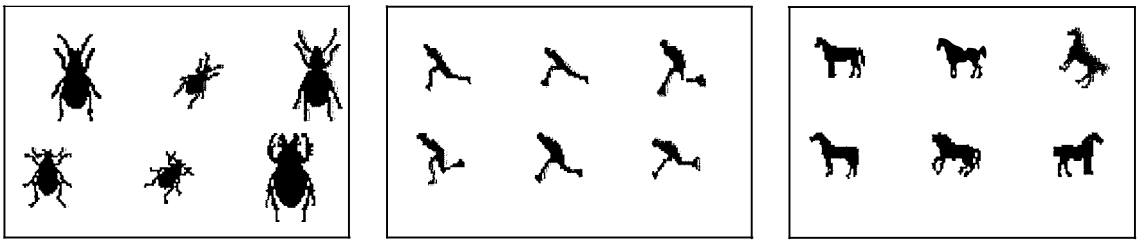
Figure VI-10 : Concept de robustesse

VI.1.2.4.3.2 - Forme basée sur le contour

Le D de forme basé sur le contour contient les caractéristiques de la forme d'un objet ou d'une région du point de vue de son contour. Ce D est basé sur la représentation de contour *Curvature Scale Space* qui possède les propriétés suivantes : elle retranscrit bien les caractéristiques des formes, elle reflète les propriétés de la perception visuelle humaine, elle est compacte, elle est robuste aux occlusions partielles des formes et aux changements de

perspectives (résultant des changements de caméra qui sont fréquents dans les images et les vidéo).

Certaines de ces propriétés sont illustrées dans les exemples ci-dessous. Chaque frame contient des images similaires du point de vue du CSS et sont les résultats de recherches effectuées dans une base de données de formes MPEG-7.



a) b) c)

a) Propriété de généralisation de la forme (similarités perceptuelles parmi des formes différentes)
b) Robustesse aux mouvements non rigides (un homme qui court)
c) Robustesse aux occlusions partielles

VI.1.2.4.4 - Localisation

VI.1.2.4.4.1 - Region Locator

Ce D permet la localisation de régions à l'intérieur d'une image ou d'une frame en les spécifiant avec une représentation brève et *scalable* d'une boîte ou d'un polygone.

VI.1.2.4.4.2 - Spatio Temporal Locator

Il décrit des régions spatio-temporelles dans une séquence vidéo (des régions d'objets en mouvement) à des fins de localisation.

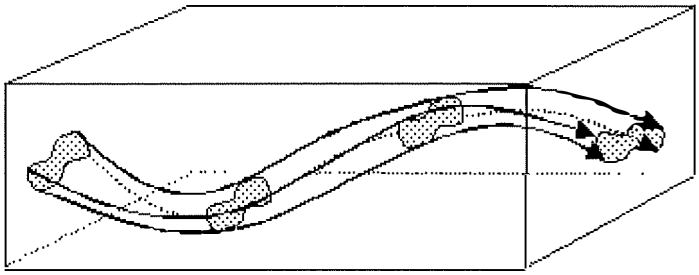


Figure VI-11 : Exemple de Spatio Temporal Locator

Comme applications à ce D, on peut citer l'hypermédia (dont le principe est de fournir des informations en rapport à un objet quand on désigne un point à l'intérieur de cet objet) et la recherche d'objets, dans quel cas on peut vérifier si un objet est passé à travers des points particuliers. Cette dernière technique peut être utilisé pour la surveillance.
Le *Spatio Temporal Locator* peut décrire des régions à la fois spatialement connectées et spatialement non connectées.

VI.1.2.5 - Entités génériques MPEG-7 et MDS

Les *schémas de description multimédia* (MDS ou DS) sont des structures de métadonnées utilisées pour décrire et annoter les contenus audiovisuels. Les DS permettent de décrire en XML et de façon standardisée les concepts importants relatifs aux descriptions de matériaux audiovisuels et à la gestion des contenus de façon à faciliter la recherche, l'indexage, le filtrage et l'accès. Les DS sont définis en utilisant le DDL.

Tandis que les D permettent de décrire des caractéristiques audiovisuelles de bas niveau telles que la couleur, la texture, le mouvement, puissance audio, etc., ainsi que certains attributs des matériaux AV tels que la location, le temps, la qualité, etc. (on s'attend d'ailleurs à ce que les caractéristiques de bas niveau soient extraites automatiquement), les DS doivent permettre de décrire des caractéristiques de plus haut niveau telles que des régions, des objets, des évènements, ainsi que des métadonnées immuables relatives à la création et à la production, à l'usage,etc. Les DS produisent des descriptions plus complexes en intégrant ensemble divers D et DS, et en déclarant les relations entre les composants de la description. Dans certains cas, des outils automatiques peuvent être utilisés pour instancier les DS, mais dans la plupart des cas une intervention humaine est requise pour la production des DS.

Dans MPEG-7, les DS sont catégorisés comme appartenant spécifiquement au domaine multimédia, audio ou vidéo. Les DS multimédia décrivent des contenus consistant en une combinaison de données audio, vidéo et textuelles, tandis que les DS audio et visuels réfèrent spécifiquement aux caractéristiques des domaines audio et vidéo.

VI.1.2.5.1 - Organisation des outils MDS

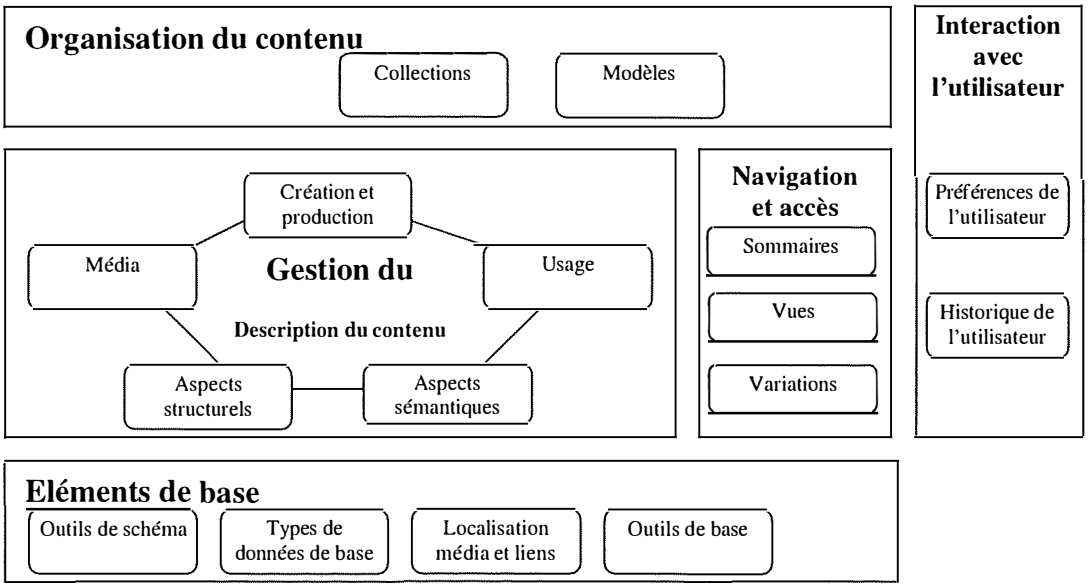


Figure VI-12 : Organisation des outils MDS

VI.1.2.5.2 - Eléments de base

VI.1.2.5.2.1 - Outils de schéma

La spécification MPEG-7 MDS a défini des *outils de schéma* qui permettent la création des descriptions MPEG-7.

- Elément Racine
  Il est le point de départ des descriptions MPEG-7. Il indique si la description est complète ou partielle. Une description complète est toujours suivie par un Elément *Top Level*. Une description partielle peut par exemple contenir des informations supplémentaires qui peuvent être ajoutées à une description existante (un D de forme, de texture, etc.). Dans le cas d'une description partielle, l'Elément Racine peut être suivi par une instance arbitraire d'un DS ou d'un D MPEG-7.

- Eléments *Top Level*
  Ils orientent la description autour d'une tâche de description spécifique, telle que la description d'un type particulier de contenu AV (par exemple une image, une vidéo, un matériau audio, ou un matériau audiovisuel) ou autour d'une fonction particulière relative à la gestion du contenu, telle que la création, l'usage, la création de sommaires, etc.

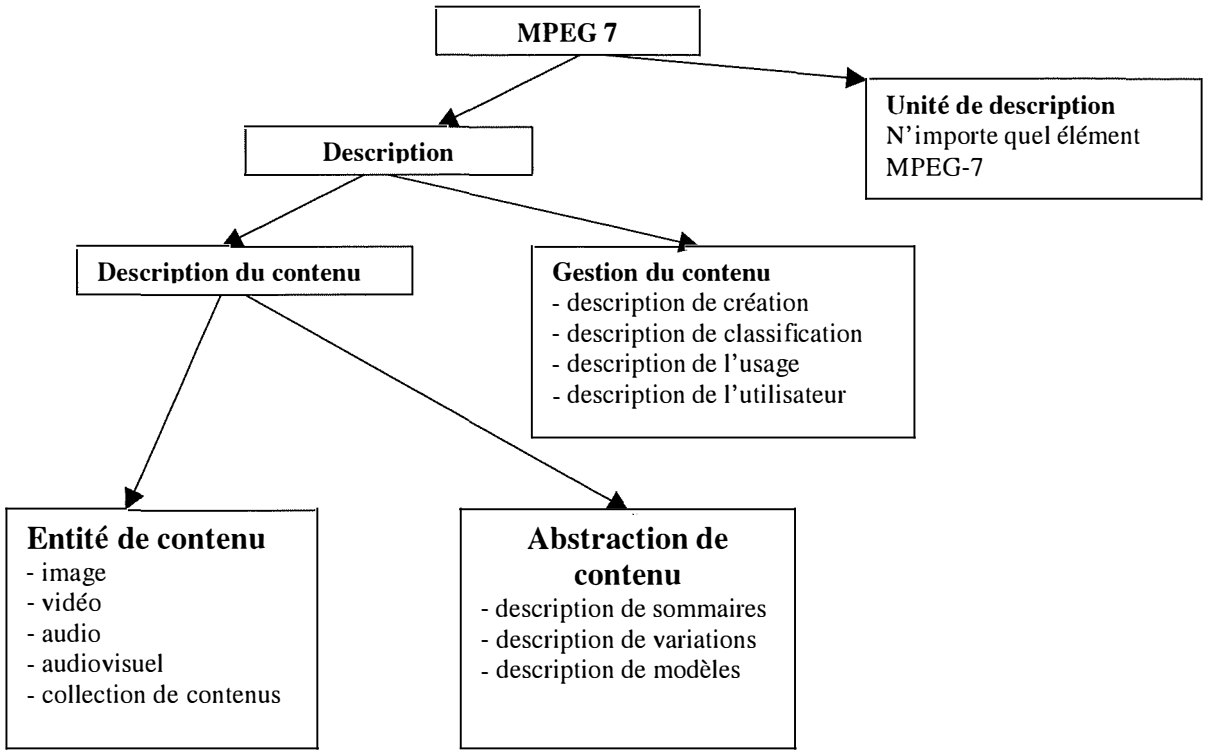


Figure VI-13 : Elément Racine d'Elément *top level*

VI.1.2.5.2.2 - Types de données de base

Les *Types de données de base* fournissent notamment des structures mathématiques telles que des vecteurs et des matrices, qui sont utilisés par les DS pour la description de contenus AV. Le *D Vecteur* peut être utilisé lorsqu'on a besoin de représenter un vecteur de nombres de taille arbitraire. Par exemple, ce D est utilisé pour représenter la valeur d'un point dans un espace de description à n dimensions. Beaucoup de D audio et vidéo l'utilisent pour représenter des points dans un espace caractéristique. Le *D Matrice* peut être utilisé lorsqu'on a besoin de représenter des matrices de nombres de taille arbitraire. Par exemple, ce D peut être utilisé pour représenter une fonction de translation entre 2 espaces de vecteurs.

VI.1.2.5.2.3 - Localisation média et liens

Les éléments de base incluent aussi des constructeurs permettant de relier des fichiers médias et de localiser des morceaux de contenu. Le *D Référence* permet de référencer une autre partie de la description. Ce D est par exemple utilisé dans les relations de graphe et dans les sommaires. Le *D Uidentificateur* permet d'identifier le contenu AV (audiovisuel) faisant l'objet d'une description. L'identificateur peut être utilisé pour identifier le contenu comme une œuvre unique ou pour identifier une de ces instances. Le *D MediaURL* utilise une URI pour localiser le contenu AV.

Exemple

<MediaURL>http://www.mpeg7.org/demo.mpg </MediaURL>
Dans cet exemple, la location d'un fichier mpg est spécifiée en utilisant une URI.

Le *DS MediaLocator* est utilisé pour spécifier la location d'une image, d'un morceau d'audio ou d'un segment vidéo particulier.

Exemple

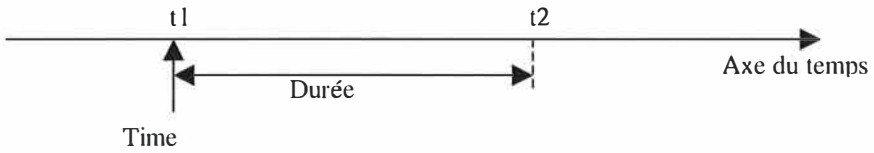
```
<MediaLocator>
    <MediaURL> http://www.mpeg7.org/demo.mpg </MediaURL>
    <MediaTime>
        <RelTime>PT3S</RelTime>
        <Duration>PT10S</Duration>
    </MediaTime>
</MediaLocator>
```

Dans cet exemple, la location d'un segment vidéo est spécifiée par l'URI d'un fichier vidéo et le moment de départ relatif par rapport au début du fichier et à la durée du segment.

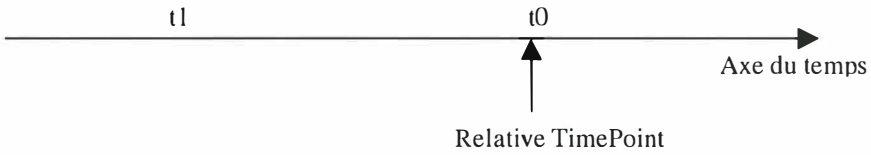
VI.1.2.5.2.4 - Outils de base

Les o*utils de base* permettent notamment de décrire des informations temporelles et d'attacher des annotations textuelles. Les *DS Time* et *DS MediaTime* décrivent les informations de temps respectivement dans le monde réel et dans les flux média. Il existe plusieurs façons de décrire un instant précis et un intervalle de temps :

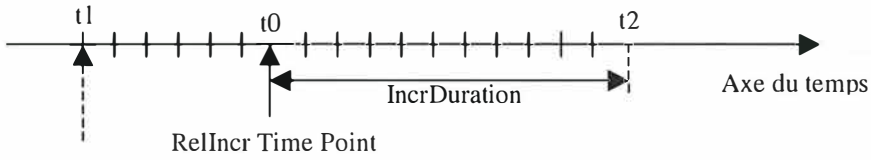
a) L'instant t1 peut être décrit par une représentation lexicale utilisant le *TimePoint*. Un intervalle [t1-t2] peut être décrit par son point de départ (t1) et sa durée (t2-t1).



b) L'instant t1 est décrit par un décalage temporel par rapport à une référence, t0, appelé *TimeBase*.



c) On peut utiliser une spécification de temps utilisant un intervalle prédéfini appelé *TimeUnit* et en comptant le nombre d'intervalles. Cette spécification est particulièrement efficace pour les signaux temporels périodiques ou échantillonnés. Le comptage des *TimeUnits* se fait par rapport à un *TimeBase*. Par exemple, dans la figure, t1 est défini par un incrément relatif de 6 *TimeUnits*. Un intervalle [t1-t2] est défini en comptant les *TimeUnits* (17).



Les annotations textuelles ont, quant à elles, une grande importance dans la construction de nombreux DS. MPEG-7 permet de construire des annotations textuelles de différentes façons: par du texte libre (mots, phrases), par des mots-clés, par du texte structuré (texte + rôle des mots), ou par des annotations de dépendances structurelles (texte structuré + relations).

Exemple

a) texte libre

```
<TextAnnotation>
    <FreeTextAnnotation  xml:lang= ''fr''>
        L'Espagne a marqué un goal contre la Suède.
        C'est Morientes qui a marqué.
    </FreeTextAnnotation>
</TextAnnotation>
```

b) mots-clé

```
<TextAnnotation>
    <KeywordAnnotation>
        <Keyword>marquer</Keyword>
        <Keyword>Suède</Keyword>
        <Keyword>Espagne</Keyword>
        <Keyword>Morientes</Keyword>
    </KeywordTextAnnotation>
</TextAnnotation>
```

c) texte structuré

```
<TextAnnotation>
    <StructuredAnnotation>
```

```
                    <Who><Name>Espagne</Name></Who>
                    <WhatAction><Name>marquer un goal </Name>
                        </WhatAction>
                    <Where><Name>Corogne, Espagne</Name></Where>
                    <When><Name>25 mars 1998</Name></When>
                </StructuredAnnotation>
            </TextAnnotation>
```

d) structure de dépendance

```
            <TextAnnotation>
                <DependencyStructure>
                  <Sentence>
                    <Phrase operator=''subject''>
                        <Head type=''noun''>Espagne </Head>
                    </Phrase>
                    <Head type=''verbe'' baseform=''marquer''>
                        marqua
                    </Head>
                    <Phrase operator=''object''>
                        <Head type=''article noun''>un goal </Head>
                    </Phrase>
                    <Phrase>
                        <Head type=''preposition''>contre </Head>
                        <Phrase>
                            <Head>Suède</Head>
                        </Phrase>
                    </Phrase>
                  </Sentence>
                </DependencyStructure>
            </TextAnnotation>
```

VI.1.2.5.3 - Gestion du contenu

Les outils décrivant la gestion du contenu permettent de décrire le cycle de vie du contenu, de la création à la consommation.

Les concepts de média maître, de profil et d'instance, qui seront utilisés dans la suite, sont expliqués ici. Le **média maître** est la source originale à partir de laquelle les différentes instances du contenu audiovisuel sont produites. Le concept de **profil** se réfère aux différentes variations qui peuvent être produites à partir d'un média original ou maître, selon les valeurs choisies pour le codage, le stockage, le format, etc. Par exemple, un concert peut être enregistré selon 2 modalités : audio et audiovisuel. Enfin, il peut exister plusieurs **instances** d'un même contenu.

Les outils de gestion du contenu sont de trois types : les outils de description du média, les outils de description de création et de production et les outils de description de l'usage.

VI.1.2.5.3.1 - Outils de description du média

Ces outils permettent de décrire le média de stockage, c'est à dire la compression, le codage, et le format de stockage des données audiovisuelles. La description du média nécessite un

élément de départ unique, le *DS Media Information*, qui identifie le média maître. Le *DS Media Information* est composé d'un D optionnel (*D Media Identification*) et d'un ou plusieurs *D Media Profile*.

D Media Identification
Le *D Media Identification* permet de décrire un contenu audiovisuel indépendamment des différentes instances disponibles.

Exemples

```
1) <MediaIdentification>
        <Identifier IdOrganization='ISO' IdName='ISBN'>0-7803-5610-1
        </Identifier>
   </MediaIdentification>

2)
   <MediaIdentification>
       <Identifier IdOrganization='MPEG' IdName='MPEG7ContentSet'>
           mpeg7_content:news1
       </Identifier>
       <VisualDomain>natural</VisualDomain>
   </MediaIdentification>
```

D Media Profile
Le *D Media Profile* contient des outils qui permettent la description d'un profil. Le profil correspondant à la copie originale du contenu audiovisuel est considéré comme le profil média maître. Pour chaque profil, il peut y avoir une ou plusieurs instances média du profil média maître.

Le *D Media Profile* est composé de :

- un **D Media Format**

  Ce descripteur contient des outils décrivant le format de codage.

  Exemple

```
<MediaFormat>
      <FileFormat>MPEG-1</FileFormat>
      <System>PAL</System>
      <Medium>CD</Medium>
      <Color>color</Color>
      <Sound>mono</Sound>
      <FileSize>666.478.608</FileSize>
      <Length>00:38</Length>
      <AudioChannels>1</AudioChannels>
      <AudioLanguage>
            <LanguageCode>es</LanguageCode>
      <CountryCode>es</CountryCode>
      </AudioLanguage>
      <AudioCoding>AC-3</AudioCoding>
</MediaFormat>
```

- un **D Media Instance**

Ce descripteur contient des outils de description qui permettent d'identifier et de localiser les différentes instances média (copies) disponibles du profil média. La location de l'instance peut être soit on-line (URL permettant un accès direct à l'instance) ou off-line.

Exemple

```
<MediaInstance>
    <Identifier IdOrganization='MPEG'IdName='MPEG7ContentSetCD'>
        mpeg7_17/news1
    </Identifier>
    <InstanceLocator>
        <MediaURL>file://D:/Mpeg7_17/news1.mpg</MediaURL>
    </InstanceLocator>
</MediaInstance>
```

- un **D Media Quality**

  Ce descripteur indique la qualité d'un contenu audiovisuel. Il peut être utilisé pour représenter à la fois la qualité subjective et la qualité objective.

U n exemple complet de MediaInformation est disponible en annexe

VI.1.2.5.3.2 - Outils de description du processus de création et de production

Ces outils permettent de décrire des informations générées par l'auteur sur le processus de création et de production du contenu audiovisuel. Ces informations ne peuvent généralement pas être extraites du contenu lui-même. Elles sont relatives au matériau mais ne sont pas explicitement indiquées dans le contenu réel. L'élément de départ de cette description est le *DS Creation Information*, qui est composé d'un *D Creation*, de 0 ou 1 *D Classification* et de 0 ou plusieurs *D RelatedMaterial.*

Il est à noter que le DS Creation Information peut être attaché à n'importe quel segment du DS Structure (voir plus loin). En effet, une description visuelle complète peut contenir des segments qui sont annotés avec plus de détails et ces segments peuvent être produits et utilisés indépendamment et/ou dans différents matériaux AV et segments.

D Creation
Ce D contient les outils de description relatifs à la création du contenu: les lieux, les dates, les actions, les matériaux, les équipes (artistiques et techniques) et les organisations impliquées.

Exemple

```
<Creation>
    <Title type="original">
        <TitleText xml:lang="es">Telediario (segunda edició
            n)
        </TitleText>
```

```
            <TitleImage>
                <MediaURL>      file://images/teledario_ori.jpg
                </MediaURL>
            </TitleImage>
        </Title>

        <Title type="alternative">
            <TitleText xml:lang="en">Afternoon news</TitleText>
            <TitleImage>
                <MediaURL>file://images/teledario_en.jpg</MediaURL>
            </TitleImage>
        </Title>

        <Creator>
            <role>presenter</role>
            <Individual>
                <GivenName>Ana</GivenName>
                <FamilyName>Blanco</FamilyName>
            </Individual>
        </Creator>

        <CreationDate>1998-06-16 </CreationDate>

        <CreationLocation>
            <PlaceName xml:lang="es">Piruli</PlaceName>
            <Country>es</Country>
            <AdministrativeUnit>Madrid</AdministrativeUnit>
        </CreationLocation>
    </Creation>
```

D Classification

Ce descripteur contient les outils de description qui permettent la classification des contenus audiovisuels. Il est utilisé à des fins de recherche et de filtrage sur base des préférences de l'utilisateur. La classification peut être orientée utilisateur (par exemple la langage, le style, le genre, etc.) ou orientée service (par exemple le sujet, l'orientation parentale, la segmentation du marché, la critique, etc.).

Exemple

```
    <Classification>
        <CountryCode>es</CountryCode>
        <Language>
            <LanguageCode>es</LanguageCode>
            <CountryCode>es</CountryCode>
        </Language>
        <Genre>News</Genre>
        <PackagedType>Information</PackagedType>
        <Purpose>broadcasting</Purpose>
        <ParentalGuidance>PG13</ParentalGuidance>
    </Classification>
```

D'autres exemples de D Classification se trouvent en annexe

D RelatedMaterial
Ce descripteur décrit les matériaux audiovisuels qui sont en corrélation avec le contenu décrit (s'il en existe).

Exemple

```
<RelatedMaterial>
      <Master>false</Master>
      <MediaType>Web</MediaType>
      <MediaLocator>
            <MediaURL>www.rtve.es</MediaURL>
      </MediaLocator>
</RelatedMaterial>
```

Un exemple complet de DS Creation Information se trouve en annexe.

VI.1.2.5.3.3 - Outils de description concernant l'usage du contenu

Ces outils décrivent les informations concernant l'usage du contenu audiovisuel. L'élément de départ de cette description est le *DS Usage Information.* Il est composé d'un *D Rights*, 0 ou 1 *D Financial* et 0 ou plusieurs *D Availability* et *D UsageRecord.* Il est intéressant de noter que cette description (par exemple le *DS UsageRecord* ou le champ *Income* dans le *D Financial*) peut évoluer chaque fois que le contenu est utilisé, ou lorsque de nouvelles façons d'accéder au contenu apparaissent, dans quel cas le *DS Availibility* doit être modifié.

D Rights
Ce D donne accès à l'information concernant le détenteur des droits sur le contenu (IPR) et aux informations concernant les droits d'accès. Le *DS Rights* fournit les références sous la forme d'identifiants uniques qui sont sous la gestion d'autorités externes. La stratégie sous-jacente est de permettre aux descriptions MPEG-7 de fournir un accès aux informations sur le détenteur des droits sans s'occuper directement des informations et de la négociation.

Exemple

```
<Rights>
      <RightsId IdOrganization='TVE' IDName='TVE_rights'>
            tve:19980618:td2
      </RightsId>
</Rights>
```

D Financial
Ce D contient l'information relative aux coûts générés et aux revenus produits par un contenu audiovisuel.

D Availability
Ce descripteur contient les outils de description relatifs à la disponibilité du contenu.

D UsageRecord
Ce descripteur fournit des informations sur l'usage passé du contenu (diffusion, distribution sur demande, vente via support CD, etc.).

Exemple

```
<UsageRecord>
     <Type>Broadcast</Type>
     <Channel>TVE:ES</Channel>
     <Place><Country>es</Country></Place>
     <Date>1998-06-16T16:30+01:00</Date>
</UsageRecord>
```

VI.1.2.5.4 - Description du contenu

MPEG-7 fournit des DS permettant de décrire la structure (segments, régions) et la sémantique (objets, événements, notions abstraites) du contenu. Il est à noter que les *DS Structure* et les *DS Semantic* sont en relation par un ensemble de liens qui permettent au contenu audiovisuel d'être décrit sur la base à la fois de sa structure et de sa sémantique. Les liens relient différents concepts sémantiques aux instances à l'intérieur du contenu audiovisuel décrit par les segments.

VI.1.2.5.4.1 - Description des aspects structurels du contenu

Les *DS Structure* permettent de décrire le contenu audiovisuel du point de vue de sa structure.

Segment

Le DS Structure est organisé autour du *DS Segment*. Un segment représente la structure spatiale, temporelle, et spatio-temporelle d'un contenu audiovisuel. Les segments peuvent ensuite être décrits par des caractéristiques perceptuelles en utilisant les D MPEG-7 de couleur, de texture, de forme, de mouvement, et les descripteurs de caractéristiques audio. Un segment peut également être décrit par des informations de création, d'usage, sur le média et par des annotations textuelles.

Le *DS Segment* est une classe abstraite (dans le sens de la programmation orientée objet). Il ne peut donc pas être instancié ; il est utilisé pour décrire les caractéristiques communes à toutes ses sous-classes.

Sous-Classes de la classe Segment

La classe abstraite *DS Segment* a 9 sous-classes :

- Un segment temporel peut être un ensemble d'échantillons dans une séquence audio, représenté par un *DS Audio Segment*, un ensemble de frames (contiguës ou non) dans une séquence vidéo, représenté par un *DS Video Segment*, ou une combinaison d'informations audio et vidéo décrites par un *DS Audio Visual Segment.*

  Exemple de *DS Video Segment* :

  Un arbre de segments vidéo peut être utilisé pour créer par exemple une table des matières. VS1 pourrait être un programme vidéo dans lequel 2 scènes, VS2 et VS4 ont été identifiées. Le segment vidéo VS1 n'est pas continu dans le temps : il est composé de 2 intervalles temporels de 6 et 3 minutes.

```
<VideoSegment id = "VS1" >
     <MediaTime>
          <MediaTimePoint> T0:0:0 </MediaTimePoint>
          <MediaDuration>PT10M</MediaDuration>
```

```
                </MediaTime>

                <MediaTimeMask NumberOfIntervals = "2">
                        <MediaTime>
                                <MediaTimePoint> T0:0:0 </MediaTimePoint>
                                <MediaDuration> PT6M </MediaDuration>
                        </MediaTime>

                        <MediaTime>
                                <MediaTimePoint> T0:7:0 </MediaTimePoint>
                                <MediaDuration> PT3M </MediaDuration>
                        </MediaTime>
                </MediaTimeMask>

                <SegmentDecomposition Gap = "true" Overlap = "true"
                DecompositionType = "temporal">
                        <VideoSegment id = "VS2" >
                                        <MediaTime>
                                          <MediaTimePoint> T0:0:0 </MediaTimePoint>
                                          <MediaDuration> PT5M </MediaDuration>
                                        </MediaTime>
                                         <GoFGoPHistogramD HistogramTypeInfo =
                                        "Average">
                                        </GoFGoPHistogramD>
                        </VideoSegment>

                        <VideoSegment id = "VS4" >
                                        <MediaTime>
                                          <MediaTimePoint> T0:0:0 </MediaTimePoint>
                                          <MediaDuration> PT6M </MediaDuration>
                                        </MediaTime>
                                        <GoFGoPHistogramD HistogramTypeInfo =
                                        "Average"> </GoFGoPHistogramD>
                        </VideoSegment>
                </SegmentDecomposition
        </VideoSegment>
```

- Un segment spatial peut être une région dans une image ou dans une frame d'une séquence vidéo. Cette région est représentée par un *DS Still Region* (qui décrit un ensemble de pixels) pour les régions 2D et par un *DS Still Region 3D* pour les régions 3D. La région spatiale peut être connectée (voir plus loin) ou non. La connectivité est indiquée par l'attribut de connectivité spatiale.

  Exemple de *DS Still Region* :

  Un arbre de régions fixes peut être utilisé pour créer une table des matières. SR1 pourrait représenter une image dans lequel deux objets, SR2 et SR3 ont été segmentés.

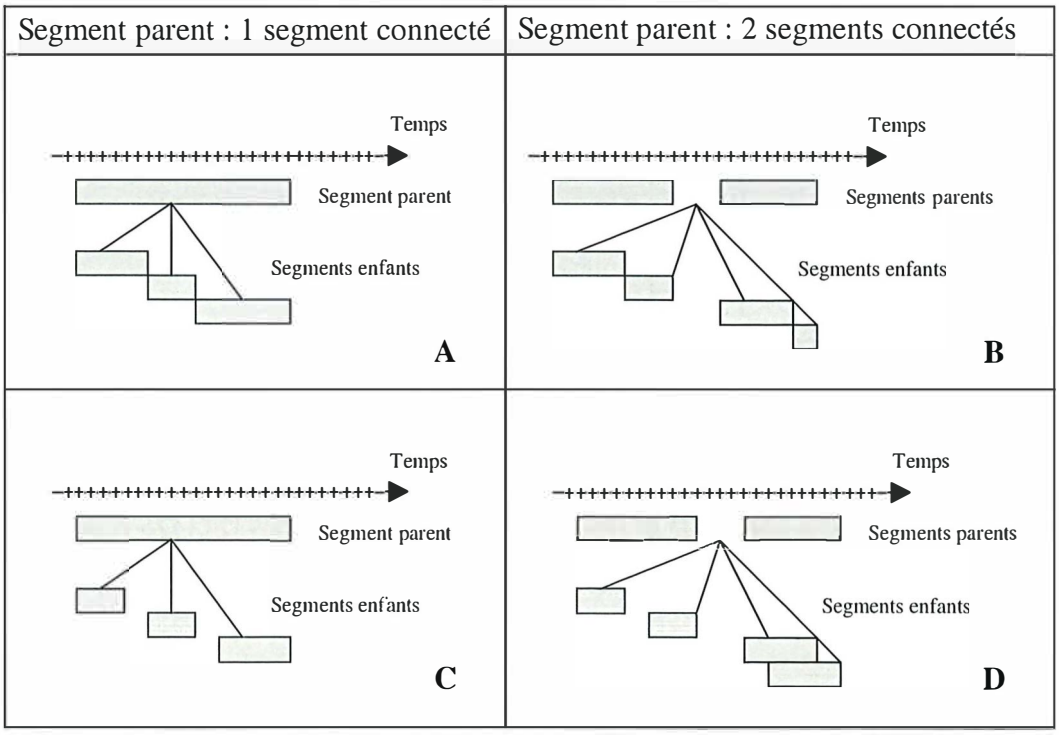
```
<StillRegion id = "SR1" SpatialConnectivity = "true">
```

```
<ContourShapeD PeakCount = "0" HighestPeak = "0">
        <GlobalCurvatureVector>
                <gcv1>2511</gcv1>
                <gcv2>8232</gcv2>
        </GlobalCurvatureVector>
</ContourShapeD>

<SegmentDecomposition Gap = "true" Overlap = "false"
DecompositionType = "spatial">
        <StillRegion id = "SR2" SpatialConnectivity "true">
                <ContourShapeD> </ContourShapeD>
        </StillRegion>

        <StillRegion id = "SR3" SpatialConnectivity = "true">
                <ContourShapeD> </ContourShapeD>
        </StillRegion>
</SegmentDecomposition>
</StillRegion>
```

- Un segment spatio-temporel peut être une région en mouvement dans une séquence vidéo (représenté par un *DS Moving Region*) ou une combinaison plus complexe d'audio et de vidéo (représenté par un *DS Audio Visual Region*).

Connecté/non-connecté

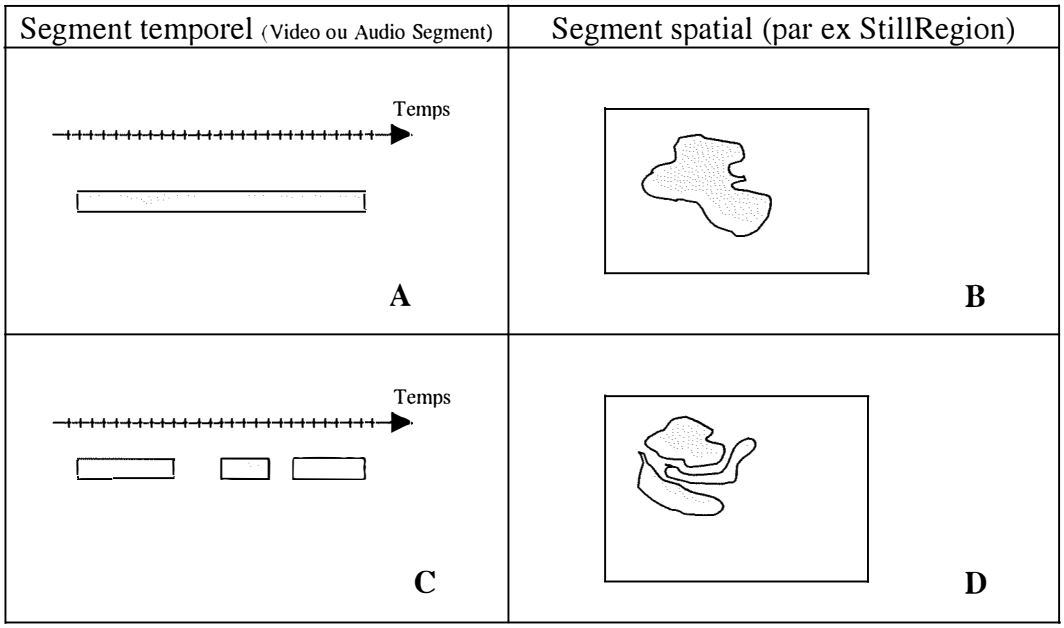
Un segment n'est pas nécessairement connecté, et peut être composé de plusieurs composants non-connectés. On parle de connectivité spatiale et de connectivité temporelle. Un segment temporel (Vidéo Segment, Audio Segment ou Audio Visual Segment) est dit temporellement connecté s'il est composé d'une séquence de frames audio ou d'échantillons audio continus. Un segment spatial (Still Region ou Still Region 3D) est dit spatialement connecté si c'est un groupe de pixels connectés.



A) Décomposition en 3 sous-segments sans trou (*gap*) ni recouvrement (*overlap*)
B) Décomposition en 4 sous-segments sans trou ni recouvrement
C) Décomposition en 3 sous-segments avec trou mais sans recouvrement
D) Décomposition en 3 sous-segments avec trou et recouvrement (un sous-segment est non-connecté)

Il est à noter que dans le dernier cas, les D et DS attachés au segment sont globaux à l'ensemble des composants connectés formant le segment. A ce niveau, il n'est pas possible de décrire individuellement les composants connectés du segment. Si les composants connectés doivent être décrits individuellement, le segment doit être décomposé en plusieurs sous-segments correspondant chacun à un composant connecté.

La décomposition doit être décrite par un ensemble d'attributs définissant le type d'une sous-division : temporel, spatial ou spatio-temporel.



A et **B** illustrent un segment spatial et temporel composé d'un composant connecté unique.
C et **D** illustrent un segment spatial et temporel composé de 3 composants connectés.

Récursivité/arbre
Le *DS Segment* est récursif, c'est à dire qu'il peut être subdivisé en sous-segments, et donc peut former une hiérarchie (arbre).

Cette décomposition hiérarchique est utile pour concevoir des stratégies de recherche efficaces (recherche globale et locale). Cela permet aussi aux descriptions d'être *scalable* : un segment peut être décrit par ses D et DS directs, ou par l'union des D et DS relatifs à ses sous-segments.

Il est à noter qu'un segment peut être divisé en sous-segments de différents types. Par exemple, un segment vidéo peut être décomposé en régions en mouvement qui sont elles-mêmes décomposées en régions statiques.

Graphe

La description de la structure du contenu par des arbres n'est pas appropriée pour certaines applications car ils impliquent certaines contraintes. Dans de tels cas, un *DS SegmentGraph* peut être utilisé. Ce DS permet de représenter des relations complexes entre les segments. La structure de graphe est définie très simplement par un ensemble de nœuds, représentant des segments, et un ensemble de liens spécifiant les relations spatio-temporelles entre les nœuds.

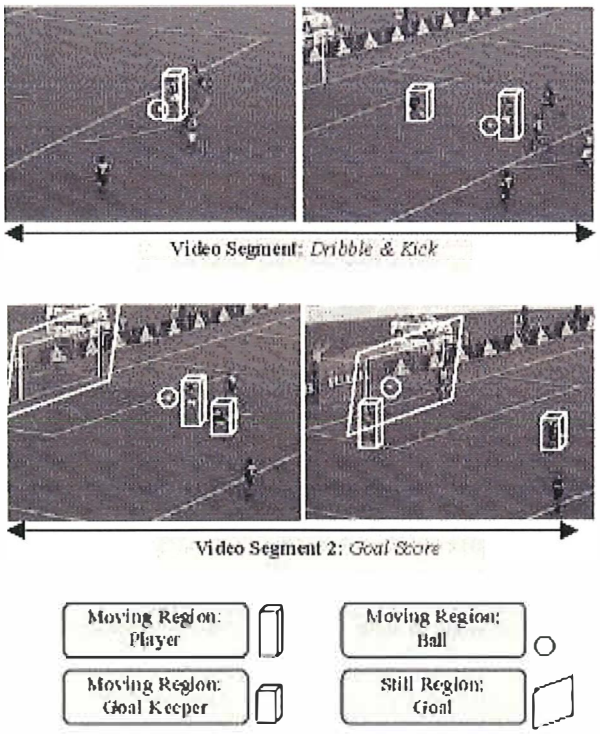


Figure VI-14 : Identification des éléments-clefs[3]

La Figure VI-14 montre un extrait d'un match de foot. Deux segments vidéo ont été définis, un *Still Région* et trois *Moving Région*. Un graphe possible décrivant la structure du contenu est montré dans la Figure VI-15.

[3] figure provenant de http://mpeg.telecomitalialab.com/standards/mpeg-7/mpeg-7.htm

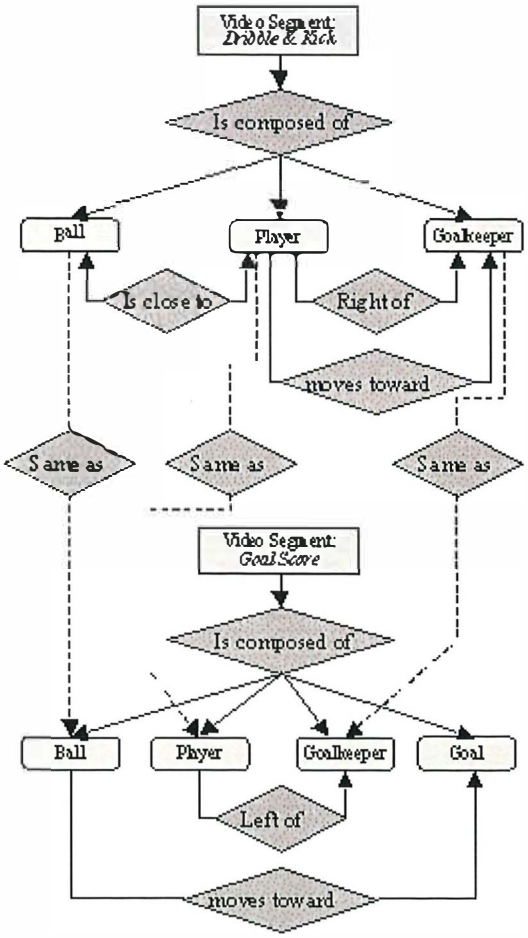


Figure VI-15 : Graphe de la structure [4]

Le segment vidéo *Dribbler et Shooter* implique la *Balle*, le *Gardien* et le *Joueur*. La *Balle* reste *proche* du *Joueur* qui se *dirige vers* le *Gardien*. Le *Joueur* apparaît à la *droite* du *Gardien*.
Le segment vidéo *Marquage du But* implique les mêmes régions en mouvement plus une région fixe appelée *But*. Dans cette partie de la séquence, le *Joueur* est à la *gauche* du *Gardien* et la *Balle* se dirige vers le *But*.
La seule information sémantique explicite est disponible grâce aux annotations textuelles (où les mots-clefs comme Balle, Joueur, ou Gardien peuvent être spécifiés).

VI.1.2.5.4.2 - Description des aspects conceptuels du contenu

Pour certaines applications, le point de vue adopté dans la section précédente n'est pas approprié. Pour les applications où la structure n'a pas de réelle utilité, mais où l'utilisateur est principalement intéressé par la sémantique du contenu, une approche alternative est fournie par le *DS Semantic*. Dans cette approche, l'accent n'est pas mis sur les segments, mais sur des *objets*, des *événements*, des *concepts*, des *endroits*, des *moments* et des *Abstraction*s.

DS

Le *DS Object* décrit un objet percevable ou un objet abstrait. Un objet percevable est une entité qui existe, c'est à dire qui a une extension temporelle et spatiale, dans un monde narratif

[4] figure provenant de http://mpeg.telecomitalialab.com/standards/mpeg-7/mpeg-7.htm

(par exemple le piano de Tom). Un objet abstrait est le résultat de l'application d'une abstraction à un objet percevable (par exemple n'importe quel piano).

Le *DS AgentObject* étend le DS Objet. Il décrit une personne, une organisation, ou groupe de gens, ou des objets personnifiés (par exemple une tasse qui parle dans un dessin animé).

Le *DS Event* décrit un événement perceptible ou abstrait. Un événement perceptible est une relation dynamique impliquant un ou plusieurs objets d'une région dans le temps et l'espace d'un monde narratif (par exemple "Tom joue du piano"). Un événement abstrait est le résultat de l'application d'une abstraction à un événement perceptible (par exemple quiconque jouant du piano).

Le *DS Concept* décrit une entité sémantique qui ne peut être décrite comme une généralisation ou une abstraction d'un objet spécifique, un événement, un moment, un lieu, ou un état. C'est l'expression d'une propriété ou d'une collection de propriétés (par exemple "l'harmonie" ou "la maturité").

Le *DS SemanticState* décrit un ou plusieurs attributs paramétriques d'une entité sémantique à un moment donné ou à une location spatiale donnée dans le monde narratif (par exemple la piano pèse 100 kg).

Le *DS SemanticTime* décrit le temps dans un monde narratif.
Le *DS SemanticPlace* décrit le lieu dans un monde narratif.

Graphe/arbre
Comme avec les segments, on peut organiser une description conceptuelle sous la forme d'un arbre ou d'un graphe. La structure de graphe est définie par un ensemble de nœuds, représentant des notions sémantiques, et un ensemble de liens spécifiant les relations entre les nœuds. Les relations sont décrites par les *DS Semantic Relation.*

Description d'abstraction
A côté de ces descriptions sémantiques d'instances individuelles, les *DS Semantic* permettent aussi la description d'abstractions. Une abstraction s'obtient par le processus suivant : on prend la description d'une instance spécifique d'un contenu audiovisuel et on la généralise à un ensemble d'instances ou à un ensemble de descriptions spécifiques.
Il existe deux types d'abstraction. Il y a tout d'abord l'abstraction média. C'est une description qui a été séparée d'une instance particulière de contenu audiovisuel, et qui peut décrire toutes les instances qui sont suffisamment similaires (la similarité dépend de l'application et du niveau de détail de la description). Un exemple typique est l'événement "nouvelles" (news), qui peut être appliqué à la description de plusieurs programmes qui peuvent avoir été diffusés sur différents canaux.
Il y a ensuite l'abstraction standard. Elle est la généralisation d'une abstraction média pour décrire une classe générale d'entités sémantiques ou de descriptions. En général, l'abstraction standard est obtenue en remplaçant les objets spécifiques, les événements ou d'autres entités sémantiques par des classes. Par exemple, si "Tom joue du piano" est remplacé par "un homme joue du piano", la description est maintenant une abstraction standard. Les abstractions standards peuvent être récursives (on peut définir des abstractions d'abstractions). Le but d'une abstraction standard est la réutilisabilité.

Exemple

La figure VI-16 montre la description des aspects conceptuels sur un exemple simple.

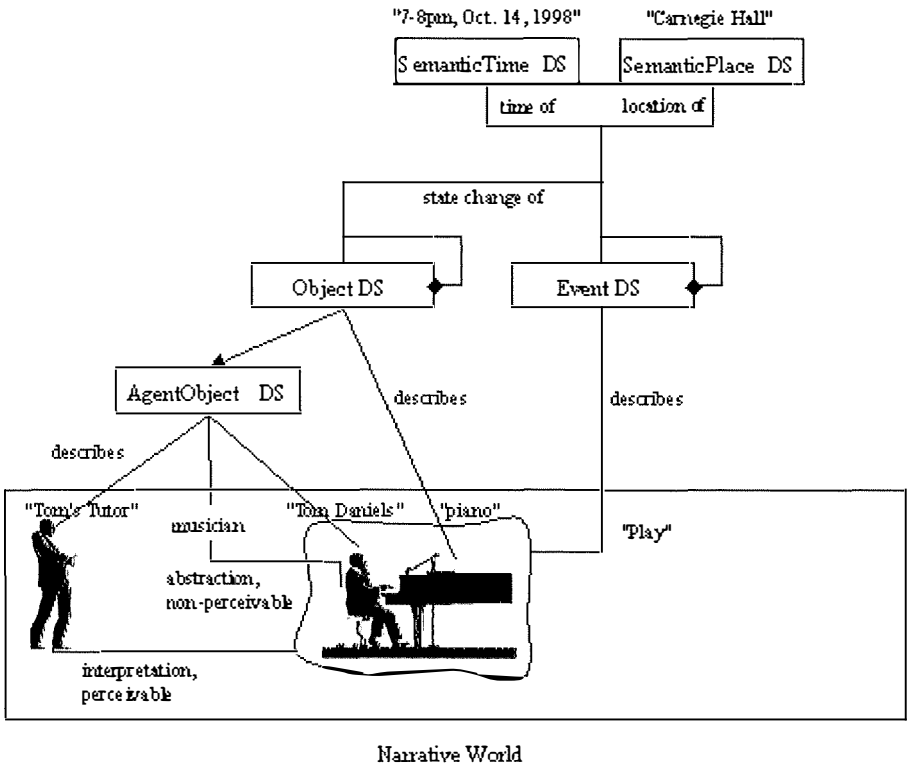


Figure VI-16 : Exemple de description des aspects conceptuels

La scène implique Tom jouant du piano et son tuteur. L'événement est caractérisé par une description sémantique du temps ("7 -8 PM le 14 octobre 1998") et un lieu sémantique ("Carnegie Hall"). La description implique un événement (jouer) et 4 objets (le piano, Tom Daniels, sont tuteur et la notion abstraite de musiciens). Les trois derniers objets appartiennent à la classe *AgentObject*.

VI.1.2.5.5 - Relations entre les descripteurs de contenu et de gestion

En pratique, la plupart des DS de description du contenu et de gestion du contenu sont reliés ensemble. Par exemple, les informations d'usage, de création et de production, et les informations médias peuvent être attachées à des segments individuels identifiés dans la description structurelle du contenu audiovisuel. D'un autre côté, selon l'application, certains aspects de la description du contenu audiovisuel peuvent être mis en évidence, telles que la description sémantique et la description de création, tandis que d'autres peuvent être minimisés ou ignorés, telles que les descriptions de média ou de structure.

VI.1.2.5.6 - Navigation et accès

Les descriptions de sommaires et de variations permettent de faciliter la navigation et l'accès aux contenus audiovisuels.

VI.1.2.5.6.1 - Sommaires

Les *DS Summarization* offrent la possibilité de créer des sommaires (compacts) du contenu audiovisuel, dans le but de faciliter la recherche et la visualisation du contenu. Les *DS Summarization* contiennent des liens vers le contenu audiovisuel, plus précisément vers les segments et les frames. Le *DS Summarization* permet la création de plusieurs sommaires du même contenu, ces sommaires pouvant être construits à différents niveaux de détails. En

utilisant des liens vers le contenu audiovisuel dans les sommaires, il est possible de générer et de stocker plusieurs sommaires sans devoir stocker plusieurs versions du contenu audiovisuel.

Il existe deux méthodes pour naviguer à travers un contenu audiovisuel. La première est la **méthode hiérarchique** (voir Figure VI-17).

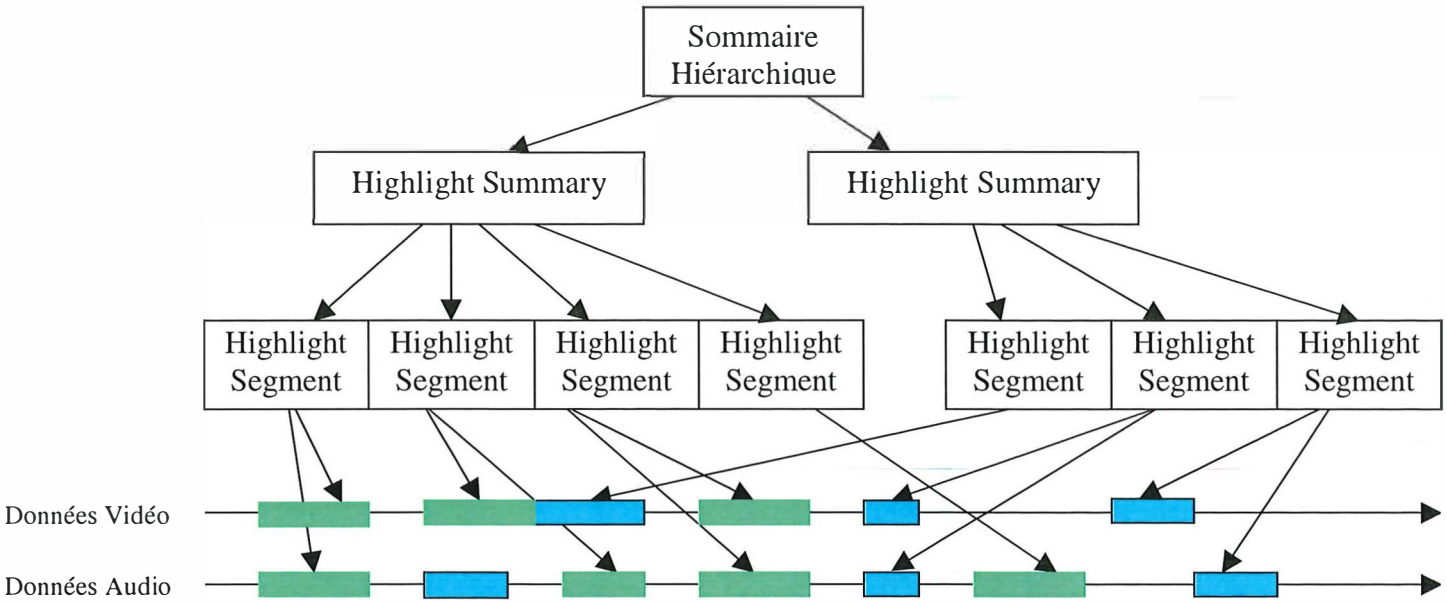


Figure VI-17 : Sommaire hiérarchique

Le *DS HierarchicalSummary* organise les sommaires en niveaux qui décrivent le contenu audiovisuel à différents niveaux de détail, de grossier à fin. En général, les niveaux proches de la racine fournissent des sommaires grossiers et les niveaux éloignés fournissent des sommaires plus détaillés. Les éléments de la hiérarchie sont spécifiés par les *DS HighlightSummary* et les *DS HighlightSegment*. La hiérarchie forme un arbre vu que chaque élément de la hiérarchie autre que la racine a un seul parent. Les éléments de la hiérarchie peuvent, éventuellement, avoir des éléments enfants.

Exemple

```
<Summarization>
    <HierarchicalSummary name="keyVideoSummary001"
    summaryTypeList="keyVideoClips"  hierarchyType="dependent">
        <RefLocator>
            <MediaURL>file://disk/video001.mpg</MediaURL>
        </RefLocator>
        <ReferenceToSegment idref="segment001"/>
        <HighlightLevel>...</HighlightLevel>
    </HierarchicalSummary>
</Summarization>
```

Le *DS HighlightSummary* est construit autour de la notion générique de segment temporel de données audiovisuelles, décrit par les *DS HighlightSegment*. Chaque *DS HighlightSegment*

contient des liens vers le contenu audiovisuel qui permettent un accès aux éléments-clés audio et vidéo, et peut aussi contenir des annotations textuelles se rapportant à des thèmes-clefs.

Exemple

La Figure VI-18 montre un exemple de sommaire hiérarchique d'une vidéo d'un match de foot.

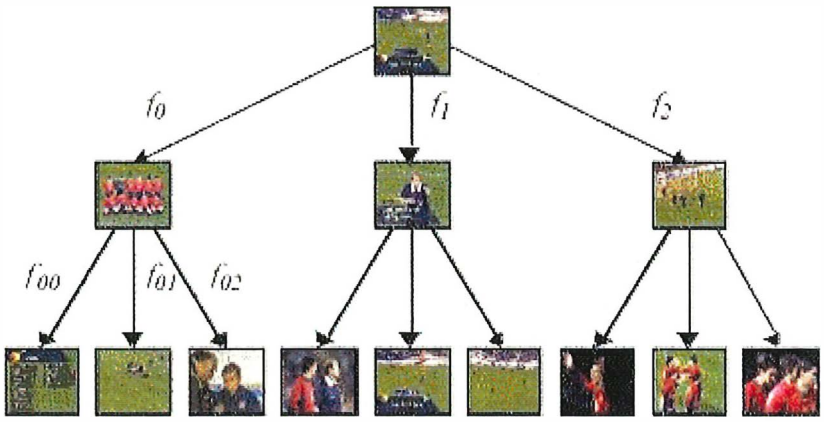


Figure VI-18 : Exemple de sommaire hiérarchique[5]

Cette description donne 3 niveaux de détails. La vidéo du jeu de foot est summarisée en une frame unique à la racine. Le niveau suivant de la hiérarchie fournit 3 frames qui summarisent différents segments de la vidéo. Enfin, le niveau le plus bas fournit des frames supplémentaires, décrivant avec plus de détails la scène dépeinte dans les segments.

La deuxième est la **méthode séquentielle**. Le *DS SequentialSummary* spécifie un sommaire consistant en une séquence d'images ou de frames vidéo, pouvant être synchronisées avec de l'audio et du texte. Ce DS peut aussi contenir une séquence de clips audio. Ce DS peut soit inclure des liens directement vers le contenu audiovisuel original afin de réduire le stockage, soit stocker séparément le contenu audiovisuel qui compose le sommaire séquentiel pour permettre une navigation et un accès rapide.

VI.1.2.5.6.2 - Variations du contenu

Les *DS Variation* fournissent des informations sur les différentes variations du contenu audiovisuel, telles que les sommaires, les versions compressées ou en basse résolution (versions *scaled*), les versions en différents langages et modalités - audio, vidéo, image, texte, etc.
Une des fonctionnalités offerte par le *DS Variation* est de permettre à un serveur, un proxy ou un terminal de sélectionner la variation adéquate du contenu audiovisuel pouvant, si nécessaire, remplacer l'original pour s'adapter aux capacités des terminaux, aux conditions réseau ou aux préférences de l'utilisateur. Une *valeur de fidélité de variation* donne la qualité de la variation comparée à l'originale.

Un attribut appelé *type* renseigne sur le type de la variation, qui peut être :

[5] figure provenant de http://mpeg.telecomitalialab.com/standards/mpeg-7/mpeg-7.htm

- Une translation : indique la conversion d'une modalité (image, vidéo, texte, audio, modèle synthétique) vers une autre.
  Des exemples de translation sont la conversion d'un texte vers de la parole, la conversion de la parole vers du texte (reconnaissance de la parole), la conversion d'une image vers du texte, …

- Un sommaire: implique une réduction des informations (suppression des détails).

- Un *scaling* : implique des opérations de transcodage des données, notamment par la réduction de la taille des données et de la qualité par la compression.

- Une extraction : implique l'extraction d'informations du programme.
  Comme exemples d'extraction, on peut citer l'extraction d'une frame-clé d'une vidéo, l'extraction d'une voix ou d'une bande audio d'un programme audio, l'extraction de texte, de régions , de segments, d'objets, et l'extraction d'événements.

- Une substitution : par exemple, un morceau de texte qui remplace une image photographique quand cette image ne peut être rendue par le terminal.

- Une révision : indique qu'un programme audiovisuel a été révisé, notamment à travers l'édition et la post-production, pour produire la variation.

VI.1.2.5.7 - Organisation du contenu

MPEG-7 propose des DS permettant d'organiser et de modéliser de collections de contenus audiovisuels, de segments, d'événements, et/ou d'objets. Ces collections permettent de décrire des propriétés communes.

VI.1.2.5.7.1 - Collections

Le *DS CollectionStructure* décrit des collections de contenus audiovisuels ou des morceaux de matériaux audiovisuels (comme des segments temporels ou vidéo par exemple). Ce DS groupe les contenus audiovisuels, les segments, les événements, ou les objets dans des grappes de collection et spécifie les propriétés communes à tous ces éléments. Il décrit aussi les relations entre ces grappes.

Exemple

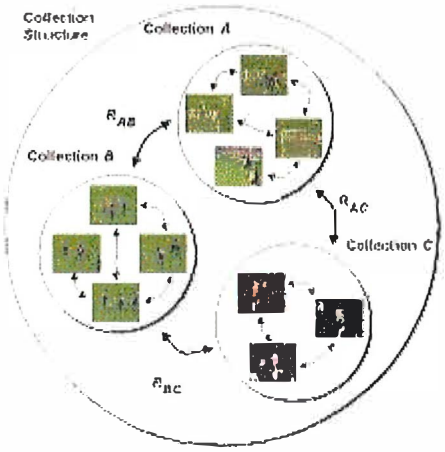


Figure VI-19 : Exemple de collection

Dans la Figure VI-19, chaque collection est constituée d'un ensemble d'images partageant des propriétés communes. Par exemple, les images décrivant un évènement similaire lors d'une rencontre de foot sont groupées ensemble. Dans chaque collection, les relations entre les images peuvent être spécifiées, tel que le degré de similarité des images à l'intérieur d'une grappe. Des relations entre les grappes peuvent également être spécifiées dans le *DS CollectionStructure*, comme par exemple le degré de similarité entre les collections.

VI.1.2.5.7.2 - Modèles

Les *DS Model* fournissent des outils de modélisation des attributs et des caractéristiques des contenus audiovisuels.

Le *DS Probability Model* permet de spécifier des fonctions statistiques et des structures probabilistes. Il peut être utilisé pour représenter des échantillons de données audiovisuelles en utilisant des approximations statistiques.

VI.1.2.5.8 - Interaction avec l'utilisateur

Le *DS User Interaction* décrit les préférences des utilisateurs (en terme de consommation du contenu audiovisuel) et l'historique d'utilisation. Les descriptions de contenu audiovisuel MPEG-7 peuvent être mises en correspondance avec les descriptions de préférence afin de sélectionner et de personnaliser le contenu audiovisuel et donc de rendre plus efficace l'accès, la présentation et la consommation.

Le *DS User Preference* décrit les préférences de l'utilisateur en termes de types de contenu et de modes de butinage. Il décrit également les dépendances contextuelles en terme de temps et de place, l'importance relative des préférences, et si les préférences sont sujettes à des mises à jour (dans le cas d'un système qui apprendrait en interagissant avec l'utilisateur).

Le *DS Usage History* décrit l'historique des actions effectuées par un utilisateur d'un système multimédia. Ces descriptions peuvent être échangées entre les consommateurs, les fournisseurs de contenu, et les appareils, et peuvent être utilisées pour déterminer les préférences de l'utilisateur par rapport à un contenu audiovisuel.

VI.2 - APPLICATIONS MPEG-7

VI.2.1 - Domaines d'application de la norme

Tous les domaines d'application qui utilisent le multimédia peuvent bénéficier de la technologie MPEG-7. Etant donné que de plus en plus de domaines sont touchés par le multimédia, les exemples ci-après n'ont de limites que celles de l'imagination. On peut citer par exemple l'architecture, les services culturels, les librairies digitales, l'e-commerce, l'éducation, le journalisme, l'édition multimédia, la surveillance, etc.

VI.2.2 - Types d'applications envisageables

Tout d'abord, il y a les applications de recherche dans une base de données, dites *pull applications,* dont voici quelques exemples de requêtes :

- En musique : jouer quelques notes au clavier et obtenir en retour une liste de morceaux contenant (ou proche de) la mélodie.
- Graphique : dessiner des lignes sur un écran et obtenir en retour un ensemble d'images contenant un graphisme ou un logo similaire.
- Scénario : décrire une action et obtenir en retour une liste de scénarios contenant une telle action.
- Voix : en utilisant un extrait de la voix de Pavarotti, obtenir une liste d'enregistrements et de clips vidéo pendant lesquels Pavarotti chante, voire même obtenir des photos de Pavarotti.

Ensuite, on a des applications de filtrage, dites *push applications.* Dans le domaine de la diffusion TV par exemple, le choix pour un utilisateur d'un canal qui l'intéresse est de plus en plus compliqué du fait de l'explosion du nombre de chaînes disponibles. On pourrait imaginer une application qui enclencherait un enregistrement par un appareil adapté lors de l'apparition d'un code dans un programme de télévision, ou encore un *reconnaisseur* d'image qui pourrait enclencher une alarme lorsqu'un certain événement visuel se produit.

MPEG-7 a également des applications potentielles dans les domaines de *la compréhension de l'image* (surveillance, vision intelligente, caméras intelligentes, etc.).

Enfin, MPEG peut faciliter la navigation (de plans en plans, de moments importants en moments importants,…), peut permettre d'obtenir des résumés audiovisuels (voir ¼ d'heure, ½ heure d'un match de foot,…) ou de personnaliser les chaînes (5 heures de programmes préenregistrés pour vous chaque jour sur les 150 canaux possibles,…), etc.

VI.2.3 - Projets en cours

VI.2.3.1 - MPEG-7 Visual Annotation Tool (application multimédia)

Cette application doit permettre aux utilisateurs de créer de manière interactive des descriptions MPEG-7 en utilisant les D et DS. L'application prend en entrée un fichier de

définition de schéma MPEG-7 (définit la structure des composants de la description MPEG-7 en utilisant le DDL) et un fichier de description de package MPEG-7 (organise les composants de la description de manière à faciliter la navigation). Dans un premier temps les données de description seront introduites à la main, mais dans le futur, seront élaborées des méthodes d'extraction automatiques et semi-automatiques des caractéristiques.

VI.2.3.2 - Customized Content Delivery for Mobile Users (application multimédia)

Cette application s'attache à transmettre des contenus multimédias à des utilisateurs mobiles. L'élément crucial dans ce cas est la flexibilité, car le temps est limité du côté du client et la masse de contenus potentiels est énorme. Dans ce contexte, on envoie des sommaires de contenus au client, sommaires qui permettront au client d'avoir une vue rapide sur le contenu afin de faire un choix éclairé.

VI.2.3.3 - Music Retrieval by Melodic Query (application audio)

Cette application doit permettre de retrouver une oeuvre musicale à partir d'un fragment de mélodie (ce sont les problèmes de *query-by-humming*). Le système utilisera le DS mélodique MPEG-7, qui incorpore le contour mélodique et des informations rythmiques (deux caractéristiques très importantes dans le processus de recherche de similarités mélodiques par l'être humain).

VI.2.3.4 - MPEG-7 Camera (application vidéo)

Le but de ce projet est de développer une caméra MPEG-7. Cette caméra sera capable d'interpréter la scène et d'extraire les informations pertinentes en temps réel. L'analyse se fait par une application logicielle. L'information extraite est alors transformée sous forme d'un flux de bits compatible MPEG-7.

VI.3 - AUTRES TRAVAUX DANS LE DOMAINE

VI.3.1 - MPEG-7 par rapport aux autres travaux dans le domaine de la description

MPEG-7 s'adresse à une multitude d'applications dans une multitude d'environnements, ce qui signifie qu'il doit fournir un cadre de travail flexible et extensible pour la description des données audiovisuelles. Pour ce faire, MPEG-7 ne définit pas un système monolithique mais plutôt un ensemble d'outils et de méthodes pour les différents points de vue de la description du contenu audiovisuel. En ayant cela en tête, MPEG-7 est conçu pour prendre en compte les points de vue considérés par les autres spécifications, parmi lesquels TV Anytime, Dublin Core, SMPTE Metadata Dictionary et UER P/Meta. Ces activités de normalisation se concentrent sur des applications ou des domaines d'application plus spécifiques, tandis que MPEG-7 a été développé aussi générique que possible. MPEG-7 est conçu pour être interopérable avec ces autres travaux. Il est à noter que MPEG entretient des rapports avec ces autres organes de normalisation.

VI.3.2 - TV-Anytime

Le forum TV-anytime, créé en septembre 1999, est une association de plus ou moins 150 organisations (fabricants d'électronique à destination des consommateurs, créateurs de contenus, opérateurs de télécommunication, diffuseurs et fournisseurs de service [Internet]). Il vise à spécifier un système qui permette au consommateur de sélectionner et d'acquérir le contenu qui l'intéresse, puis de le regarder quand il le veut.
Le cadre des activités du forum repose sur un modèle de données et sur un format commun de représentation des métadonnées, ce qui permet de reprendre les outils de métadonnées développés par d'autres groupes (MPEG-7, SMPTE, P/META par exemple) en plus de ceux définis par le forum.
Avec la solution TV-Anytime, des applications pourront exploiter la mémoire locale de plates-formes grand public. L'utilisateur pourra parcourir, sélectionner et acquérir des contenus indépendamment de leur système de distribution. Le forum travaille à la rédaction de spécifications de systèmes ouverts, compatibles, intégrés et sécurisés, reliant les créateurs et fournisseurs de contenu aux consommateurs, via les fournisseurs de services.

L'objectif de TV-Anytime est de permettre la recherche, le filtrage, la localisation et l'acquisition de contenu, peut importe où il se trouve et les conditions de sa disponibilité.

Lorsqu'il utilise le système de métadonnées de TV-Anytime, le consommateur reçoit des informations descriptives pendant les phases de recherche, de sélection, de navigation et de gestion du contenu. Les *attracteurs* sont les informations que le téléspectateur ou le dispositif automatique utilise pour décider ou non d'acquérir un contenu particulier. Le système pourra utiliser des profils ou indications de préférence d'utilisateur pour aider le téléspectateur dans cette tâche à l'aide par exemple d'un dispositif utilisant la technologie des agents intelligents.

Le système de référence du contenu est un processus logique qu prend place entre la fin du processus de détection et l'acquisition du sujet. Les mécanismes mis en œuvre satisferont à la demande en associant un ou plusieurs *localisateurs* ou systèmes de repérage à l'identificateur du contenu lui-même.

Des règles définiront au cas par cas l'accès au contenu et aux métadonnées. La gestion des droits relatifs au contenu est essentielle à leur protection. TV-Anytime devra également sécuriser son système afin de protéger les métadonnées telles que les informations descriptives, qu'elles soient privées ou relevant de la gestion des droits.

Les groupes de métadonnées définis sont la description du programme, la description de l'instance, d'autres métadonnées relatives au CRID (identificateur de référencement du contenu), et des métadonnéees sur le consommateur.

Pour des raisons d'interopérabilité (le processus de création et d'acheminement vers le consommateur des métadonnées pour un contenu peut impliquer plusieurs organisations), le forum a adopté XML comme format de représentation pour les métadonnées. TV-Anytime a opté pour le langage de description MPEG-7 (le DDL) pour décrire la structure des métadonnées et l'encodage XML des métadonnées. Les métadonnées peuvent être encodées dans un format binaire pour la transmission ou le stockage.

VI.3.3 - SMPTE (Society of Motion Pictures and Television Engineers)

SMTPE publie des normes approuvées par l'ANSI, des pratiques recommandées et des lignes directrices dans le domaine de l'ingénierie. Aujourd'hui, SMPTE est reconnu comme le leader mondial dans le développement de normes et de pratiques faisant preuve d'autorité dans les domaines du film, de la télévision, de la vidéo et du multimédia.
La EBU/SMPTE Task Force a été initiée en 1996 au constat que l'explosion prochaine de l'utilisation des technologies de production digitale pour la télévision allait engendrer des problèmes d'interopérabilité. L'objectif de cette Task Force est l'établissement de méthodes efficaces et interopérables pour l'échange de matériaux de programmes de télévision entre les systèmes.

Plus spécifiquement, SMPTE a développé la spécification EPG (Electronic Program Guide).
Vous pourrez trouver de plus amples informations à l'adresse suivante :
http://www.3veni.com/epg.pdf.

VI.3.4 - UER P/META

Les travaux du groupe P/META ont une double origine : le groupe d'action SMPTE/UER (Union Européenne de Radio-Télévision) et le projet *Media Data* de la BBC.
Le système P/META vise les échanges de contenu entre des entités professionnelles, à savoir les secteurs de la production (ou fournisseur de contenu, responsable de la création d'un programme complet), de la distribution (chargé de réunir les programmes et de les faire parvenir aux consommateurs) et des archives (chargé de l'archivage, de la récupération, de la gestion d'accès et de la conservation du contenu). Le système P/META se concentre sur les trois entités professionnelles, d'autres organismes s'occupant de définir les métadonnées pour les consommateurs, notamment le forum TV-Anytime.
Les transactions entre entreprises requièrent des informations qui identifient ou fournissent le matériel, les données rédactionnelles et descriptives sur le matériel, le droit d'utiliser le matériel et le format du matériel, afin de s'assurer que le récepteur sera capable de l'utiliser correctement.

Le système P/META comprend les documents suivants : un tableau de définition des attributs, un tableau des valeurs des attributs (lorsqu'un attribut peut prendre des valeurs contrôlées), des définitions d'ensembles d'attributs (regroupements logiques ou structurés d'informations nécessaires pour prendre en charge certaines transactions) et des scénarios et exemples (servant à tester et à valider la qualité de la prise en charge assurée par le système P/META pour les transactions).

L'intérêt de définir une relation de correspondance entre les systèmes P/META et TV-Anytime est évident. On pourrait ainsi mieux satisfaire les transactions d'échange avec le quatrième secteur (le consommateur).

VI.3.5 - DUBLIN CORE

Dublin Core Metadata Initiative (DCMI)

Le DCMI est une organisation dont le but est de promouvoir une large adoption d'un standard de métadonnées interopérable et de développer un vocabulaire de métadonnées spécialisé pour la description de ressources afin de rendre les systèmes de recherche plus performants.

La mission de DCMI est de rendre la recherche de ressources plus facile par le développement des activités suivantes :

- le développement de normes de métadonnées pour la recherche dans des domaines variés
- la définition d'un cadre de travail pour l'interopération des ensembles de métadonnées
- la facilitation du développement d'ensembles de données spécifiques à une communauté ou à une discipline qui est compatible avec les points 1 et 2.

Dublin Core Metadata Element Set (DCMES)

Le DCMES est un standard simple et extensible de métadonnées dont le but est de promouvoir la recherche de ressources sur le web.
Dans ce langage, il y a deux classes de termes : les éléments (noms) et les attributs (adjectifs). De par ses caractéristiques, le DCMES peut être facilement maîtrisé mais il ne permet pas d'exprimer des relations ou des concepts complexes. C'est pourquoi il doit coexister avec d'autres standards de métadonnées.

DCMES défini des éléments de contenu (la couverture, la description, le type, la relation, la source, le sujet, le titre), des éléments de propriété intellectuelle (le contributeur, le créateur, le publieur, les droits) et des éléments d'*instantiation* (la date, le format, l'identificateur, le langage).

La simplicité de DCMES permet de réduire les coûts de création des métadonnées et encourage l'interopérabilité.

PARTIE 7

CONCLUSION

VII - CONCLUSION

On peut remarquer l'ambition de ces normes à ratisser large et une certaine ouverture permettant des évolutions futures.

MPEG-2 fut créé car MPEG-1 était par trop limitatif, MPEG-4 n'avait par vraiment pour but de remplacer le MPEG-2 mais s'orientait plutôt vers les bas débits avant que son objectif ne soit considérablement agrandi et qu'il ouvre une nouvelle voie de compression grâce à la gestion des objets vidéos.

Qu'en est-il de l'avenir de ces normes ? Est-ce que les évolutions technologiques dans le domaine des réseaux ne risquent-elles pas de rendre les techniques de compression dispensables ? Il n'y a pas de doute là-dessus : l'appétit des utilisateurs est toujours grandissant…De plus les techniques à large bande que nous connaissons actuellement dans nos pays (de type ADSL), bien que représentant une révolution, ne sont en fait qu'un moyen d'utiliser au maximum les infrastructures existantes. La solution technologiquement parfaite et ouvrant de réelles nouvelles perspectives, à savoir la fibre optique du fournisseur au consommateur, représente de gros investissements qui ne seront pas consentis dans un avenir proche. Dès lors ces techniques de compression sont toujours indispensables et sûrement insuffisantes pour satisfaire l'engouement des consommateurs en termes de multimédia et d'applications diverses.

Quant à MPEG-7, il jette les bases d'un langage universel de sémantique des contenus transportés à travers les réseaux. On réalise la portée de cette norme quand on constate le nombre de projets qui ont déjà été entrepris alors que la norme n'était pas encore à son stade final. Nul doute que ce langage, opérationnel et complet, sera à la base des développements technologiques futurs, nombreux et variés.

Enfin, l'interopérabilité semble être le maître mot d'un développement technologique durable et profitable. En somme de nombreux travaux ont déjà été réalisés, mais les festivités ne font que commencer.

Partie 8

Bibliographie

VIII - BIBLIOGRAPHIE

VIII.1 - SOURCES NORMALISATION

http://www.iso.ch
http://www.cselt.it
http://www.iec.ch

« Normalisation et standardisation dans les nouvelles technologies »
Jean-Alain Hernandez – Ecole nationale supérieure de telecommunications Direction de la formation continue

VIII.2 - SOURCES MPEG-1

« **A la recherche des images et des sons** » *Leonardo Chiariglione*
(dernière lecture le 21/05/2002)
http://leonardo.telecomitalialab.com/paper/ina_fr97/ina_fr97.html

« **MPEG Video Homepage** » *Heinrich-Hertz-Institut Berlin – Image Processing Department, Thomas Sikora*
(dernière lecture le 12/02/2002)
http://wwwam.hhi.de/mpeg-video/

« **Pulsent** »
(dernière lecture le 21/05/2002)
http://www.pulsent.com/solutions/applications.html

« A Rule of Thunm for Comparing Video Compression »
Robert F. Rice – Pulsent Corporation (9 novembre 2001)

« Compressions MPEG-1 à MPEG-4 »
Etienne FERT – Responsable de la division Traitement numérique du signal aux laboratoires d'électronique Philipd et Sylvie Jeannin – Ingenieur de recherche aux laboratoires d'électronique Philips.

« Les Formats Vidéo Numériques »
Philippe GASSER – Service des technologies de l'audiovisuel et de la communication éducative
(dernière lecture le 21/05/2002)
http://www.cndp.fr/notestech/27/nt027_0.htm

« A la recherche des images et des sons »
Leonardo Chiariglione, Président du groupe MPEG
(dernière lecture le 21/05/2002)
http://leonardo.telecomitalialab.com/paper/ina_fr97/ina_fr97.html

« Introcuction – Définitions – MPEG Systems – MPEG Audio … »
Pscart et Van Muysewinkel Olivier – I.S.I.B. – Ingénieur Industriel en Electronique (section Réseaux Informatiques)
(dernière lecture le 21/05/2002)
www.isib.be

« The Ins and Outs of MPEG »
Elizabeth Cheesbrough
(dernière lecture le 21/05/2002)
http://www.rcc.ryerson.ca/rta/brd038/papers/1996/mpeg1.htm

« Video and Audio Compression »
(dernière lecture le 21/05/2002)
http://www.cs.sfu.ca/CourseCentral/365/li/material/notes/Chap4/Chap4.html

« Mpeg-1 and Mpeg-2 Digital Video Coding Standards »
Thomas Sikora - Heinrich-Hertz-Intitut Berlin – Image Processing Department

« Video Compression Demystified »
Peter Symes – Editions Mc GrawHill
http://books.mcgraw-hill.com/cgi-bin/pbg/0071363246.html

http://iphilgood.chez.tiscali.fr
(dernière lecture le 21/05/2002)

www.media-video.com
(dernière lecture le 21/05/2002)

www.labdv.com
(plus disponible sans abonnement)

www.planete-numerique.com
(dernière lecture le 21/05/2002)

VIII.3 - SOURCES MPEG-2

Concernant MPEG :

- http://www.mpeg.org/MPEG/index.html (consulté fin 2001)

Concernant l'analyse technique :

- **« ATM & MPEG-2: Integrating Digital Video Into Broadband Networks »**
  Michael Orzessek et Peter Sommer, – Editions Prentice Hall PTR – 1997(First Editoin)
- http://mpeg.telecomitalialab.com/standards/mpeg-2/mpeg-2.htm (consulté fin 2001)
- http://mpeg.telecomitalialab.com/documents/dsmcc/dsmcc.htm (consulté fin 2001)

Concernant le DVD :

- http://www.mpeg.org/MPEG/DVD (consulté en mars 2002)

- http://www.dvddemystified.com/dvdfaq.html (consulté en mars 2002)

Concernant le DVB :
- http://www.erg.abdn.ac.uk/public_html/research/future-net/digital-video/dvb.html (consulté février 2002)
- http://iphilgood.chez.tiscali.fr/ (consulté en mars 2002)
- http://www.mines.u-nancy.fr/~tisseran/I33_Reseaux/compression.video/dvb.htm (consulté février 2002)

Concernant Pulsent :
- http://www.clubic.com/n/n5471.html (consulté en mars 2002)
- http://www.pulsent.com/press/pressrelease032502.html (consulté en mars 2002)
- http://www.internetnews.com/infra/article/0,,10693_997141,00.html (consulté en mars 2002

VIII.4 - SOURCES MPEG-4

« Overview of the MPEG-4 Standard »
Rob Koenen – Mars 2001

« Video Compression Demystified »
Peter Symes – Editions Mc GrawHill – Parution en 2001

« Dossiers MPEG 4.0 »
(dernière lecture le 09/05/2002)
http://www.3d-test.com/dossiers/dossier-mpeg-1.htm

« La compression vidéo MPEG »
GERBER Jacques-Alexandre et GIGNOUX Sébastien
Exposés des élèves du cours de deuxième année de l'Ecole des Mines de Nancy RESEAUX 1996/1997
(dernière lecture le 09/05/2002)
http://www.mines.u-nancy.fr/~tisseran/I33_Reseaux/compression.video/mpeg4.htm

« Présentation de MPEG4 »
Benoit Luttriger
(dernière lecture le 09/05/2002)
http://membres.lycos.fr/benoitluttringer/Annexe_Mpeg4.html

VIII.5 - SOURCES MPEG-7

Analyse technique MPEG-7 :
- http://www.expway.tv/tutorial/mpeg7bim/powerpoint.html (consulté en février 2002)
- http://www.expway.tv/tutorial/mpeg7system/powerpoint.html
- (consulté en février 2002)
- http://www.mpeg-industry.com/ifg/IBC_BiM_2001.ppt (consulté début 2002)
- http://www.onlinemag.net/OL2001/day9_01.html
- http://mpeg.telecomitalialab.com/standards/mpeg-7/mpeg-7.htm

(consulté fin 2001 et a connu des modifications importantes début 2002)
- http://www.knk-mpeg.com/mpegnews.htm (consulté début 2002)
- http://www.revue-i3.org/Seminaires/Transparents_Philippe_Joly.pdf (consulté en avril 2002)

XML :
- http://madsemusdipc1.insa-rouen.fr/tutoriaux/xml (consulté fin 2001)
- http://www.commentcamarche.net/xml/xmlintro.php3 (consulté fin 2002)
- http://www.chez.com/xml/initiation/index.htm (consulté fin 2002)
- http://www.w3.org/TR/xmlschema-0/ (consulté début 2002)
- http://www.w3.org/TR/xmlschema-1/ (consulté début 2002)
- http://www.w3.org/TR/xmlschema-2/ (consulté début 2002)

TV-Anytime :
- http://www.ebu.ch/trev_284-evain_fr.pdf (consulté en avril 2002)
- http://xml.coverpages.org/tvAnytime.html (consulté en avril 2002)
- http://xml.coverpages.org/TVAnytime-SP003v11.pdf (consulté en avril 2002)
- http://www.3veni.com/epg.pdf (consulté en avril 2002)

UER P/META :
- http://www.ebu.ch/trev_284-hopper_fr.pdf (consulté en avril 2002)

Dublin Core :
- http://dublincore.org/documents (consulté en avril 2002)
- http://dublincore.org/documents/1999/07/02/dces/ (consulté en avril 2002)

PARTIE 9

ANNEXES

IX - ANNEXES

IX.1 - NORMALISATION

IX.1.1 - Stades de l'élaboration des Normes internationales

Une Norme internationale est le résultat d'un accord entre les comités membres de l'ISO. Elle peut être employée telle quelle ou peut être mise en oeuvre par voie d'incorporation dans les normes nationales des différents pays.

Les Normes internationales sont élaborées par les comités techniques (TC) et sous-comités (SC) de l'ISO selon un processus qui comporte six étapes:
Stade 1: Stade proposition
Stade 2: Stade préparatoire
Stade 3: Stade comité
Stade 4: Stade enquête
Stade 5: Stade approbation
Stade 6: Stade publication

Si un document possédant un certain degré de maturité est disponible dès l'amorce d'un projet de normalisation, par exemple une norme élaborée par une autre organisation, il est possible d'omettre certains stades. Dans le cadre de la "Procédure par voie expresse", un document est soumis directement pour approbation en tant que projet de Norme internationale (DIS) aux comités membres de l'ISO (stade 4) ou, si le document a été élaboré par un organisme international à activités normatives reconnu par le Conseil, en tant que projet final de Norme Internationale (FDIS, stade 5), sans passer par les stades précédents.

Un résumé de chacun des six stades concernés est donné ci-dessous.

IX.1.1.1 - Stade 1: Stade proposition

La première étape de l'élaboration d'une Norme internationale consiste à confirmer qu'il existe un besoin pour la Norme internationale en question. Une demande de mise à l'étude d'une nouvelle question (NP) est soumise au vote des membres du TC/SC concerné afin de décider s'il y a lieu d'inscrire la question au programme de travail.
La demande est acceptée si la majorité des membres (P) du TC/SC se prononce en sa faveur et qu'au moins cinq membres (P) s'engagent explicitement à participer activement au projet. Normalement, à ce stade, un chef de projet est désigné pour prendre la direction de l'étude.

IX.1.1.2 - Stade 2: Stade préparatoire

En général, un groupe de travail composé d'experts, dont le président (animateur) est le chef de projet, est mis en place par le TC/SC pour préparer un avant-projet. Plusieurs avant- projets successifs peuvent être examinés jusqu'à ce que le groupe de travail ait acquis la certitude

d'avoir élaboré la meilleure solution technique au problème considéré. A ce stade, le projet est transmis au comité responsable du groupe de travail pour aborder la phase de recherche de consensus.

IX.1.1.3 - Stade 3: Stade comité

Dès qu'un premier projet de comité (CD) est disponible, celui-ci est enregistré au Secrétariat central de l'ISO. Il est diffusé pour observations, voire pour vote, auprès des membres (P) du TC/SC. Plusieurs CD successifs peuvent être examinés jusqu'à ce qu'un consensus soit atteint sur le contenu technique du document. Une fois ce consensus obtenu, il est procédé à la mise au point définitive du texte en vue de sa soumission comme projet de Norme internationale (DIS).

IX.1.1.4 - Stade 4: Stade enquête

Le projet de Norme internationale (DIS) est distribué à tous les comités membres de l'ISO par le Secrétariat central de l'ISO pour vote et observations dans un délai de cinq mois. Il est approuvé en tant que projet final de Norme internationale (FDIS) si une majorité des deux tiers des membres (P) du TC/SC se prononce en sa faveur et qu'au plus le quart de l'ensemble des voix exprimées est défavorable. Si les critères d'approbation ne sont pas remplis, le texte est renvoyé au TC/SC d'origine pour étude complémentaire et un document révisé est à nouveau distribué pour vote et observations en tant que projet de Norme internationale.

IX.1.1.5 - Stade 5: Stade approbation

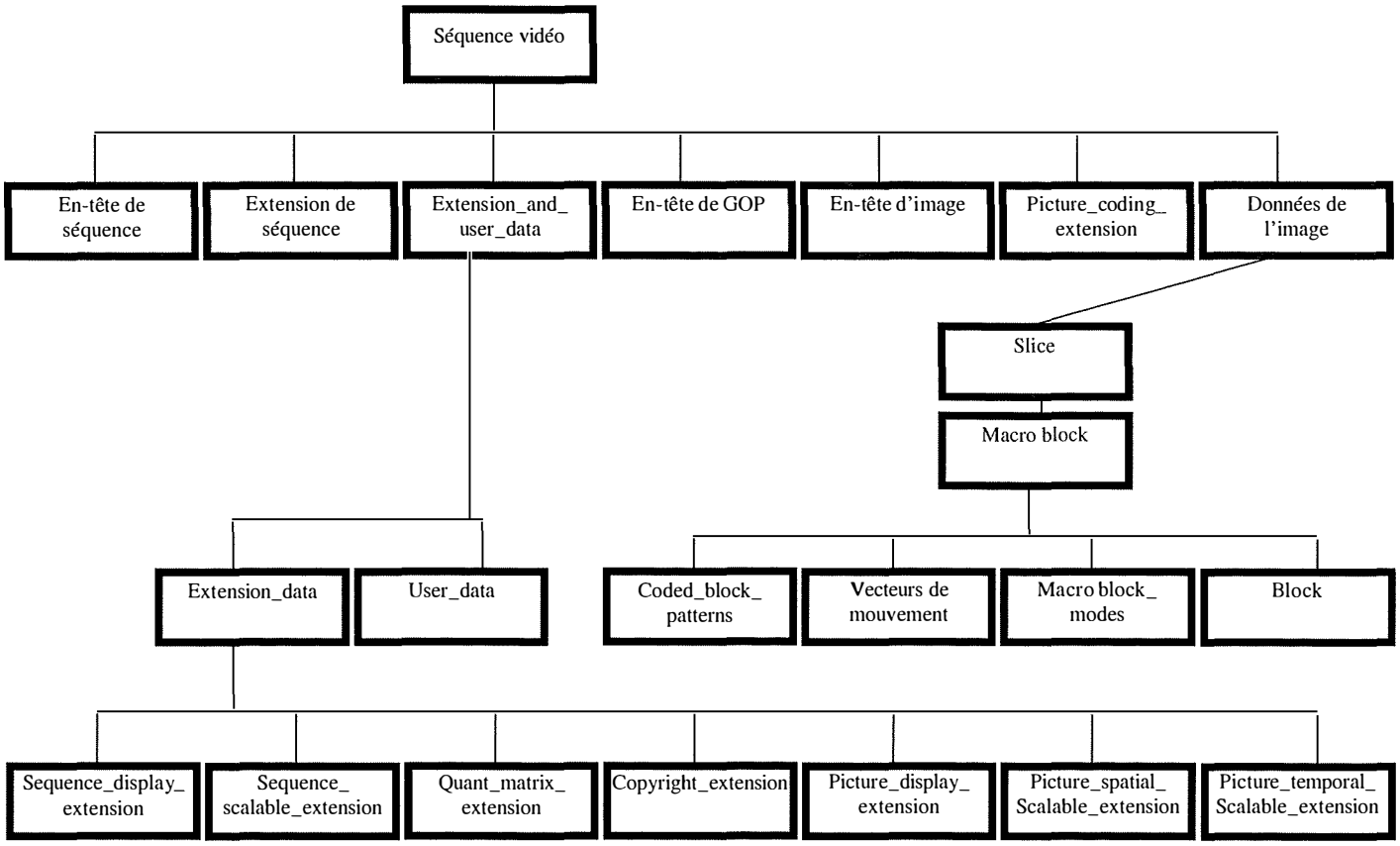
Le projet final de Norme internationale (FDIS) est distribué à tous les comités membres de l'ISO par le Secrétariat central de l'ISO pour vote final par Oui ou par Non dans un délai de deux mois. Si des observations techniques sont recueillies durant ce délai, elles ne sont pas prises en compte à ce stade, mais sont consignées pour examen lors d'une révision ultérieure de la Norme internationale. Le texte est approuvé en tant que Norme internationale si une majorité des deux tiers des membres (P) du TC/SC se prononce en sa faveur et qu'au plus le quart de l'ensemble des voix exprimées est défavorable. Si les critères d'approbation ne sont pas remplis, le texte est renvoyé au TC/SC d'origine pour étude complémentaire à la lumière des arguments techniques présentés à l'appui des votes négatifs recueillis.

IX.1.1.6 - Stade 6: Stade publication

Lorsque l'approbation d'un projet final de Norme internationale est acquise, seules des modifications mineures, d'ordre rédactionnel, sont apportées au texte final, lorsque cela s'impose. Le texte définitif est envoyé au Secrétariat central de l'ISO, qui procède à la publication de la Norme internationale.
Examen périodique des Normes internationales (confirmation, révision, annulation)
Toutes les Normes internationales sont réexaminées au moins une fois tous les cinq ans par le TC/SC responsable. Il est décidé à la majorité des membres (P) du TC/SC si une Norme internationale doit être confirmée, révisée ou annulée.

IX.2 - MPEG-2

IX.2.1 - Syntaxe des flux de bits vidéo MPEG-2 :



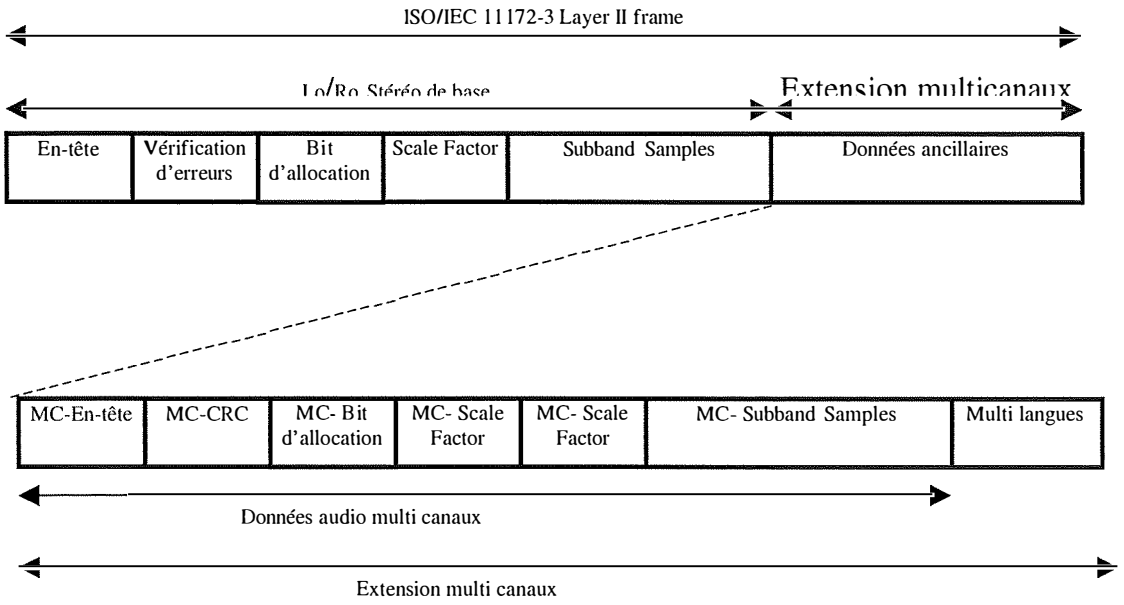
La **Séquence Vidéo** est une structure contenant des images et des données d'extension.
L'**En-tête de séquence** contient les informations concernant la taille de l'image et le nombre d'images par seconde. Ces informations sont utilisées si le décodeur doit synchroniser un nouveau programme lors d'un changement de flux vidéo.
L'**Extension de séquence** contient le profil, le niveau et le format de chrominance du flux de bits. Si ce champ n'est pas présent, le flux de bits est un flux MPEG-1 Vidéo.

L'**En-tête de GOP** contient un timecode et des informations sur les images formant le GOP.
L'**En-tête d'image** indique si l'image est de type I, P, ou B. Il contient également un champ de référence temporel, qui indique 'the display sequence'.
Le **Picture_Coding_Extension** contient des informations sur les images afin de supporter les modes interlace/progressive et les standards vidéos analogiques spécifiques (NTSC, PAL).
Le champ **Données de l'image** contient plusieurs slices.
Un **Slice** contient la position verticale du slice, l'information concernant le partitionnement de données, l'échelle de quantification, et plusieurs structures de macro blocs.

Un **Macro bloc** contient une échelle de quantification optionnelle, des blocs, un macroblocks_modes, et des vecteurs de mouvement.
Le **Mode macro bloc** indique la façon dont le macro bloc est encodé. Dans une image prédite, un macro bloc peut être encodé comme un macro bloc intra ou comme un macro bloc prédit. Dans une image bidirectionnelle, un macro bloc peut être encodé comme un macro bloc intra, prédit ou bidirectionnel. Ce champ indique également le mode de prédiction utilisé pour le macro block bidirectionnel.
Le champ **Vecteurs de mouvement** contient les vecteurs de mouvement pour un macro bloc donné.
Le **Coded_Block_Pattern** indique en fait quels sont les blocs du macro bloc qui sont réellement encodés. Dans le cas où tous les coefficients d'un bloc valent zéro après la quantification, ce bloc n'a pas besoin d'être encodé (dans un macrobloc qui est utilisé dans une P- ou B-picture).
Un **Block** contient des coefficients DCT.

Le **User_data** contient des données définies par l'utilisateur.
Le **Sequence_Display_Extension** contient des informations supplémentaires sur le format vidéo et les attributs de couleur utilisés dans le flux de bits.
Le **Sequence_Scalable_Extension** indique modes de scalabilité utilisés dans le flux de bits. Il fournit également des informations au décodeur sur la manière de manipuler la scalabilité.
Le **Quant_Matrix_Extension** contient des matrices définies par l'utilisateur pour la quantification inverse.
Le **Picture_Display_Extension** contient des informations utilisées lors du processus de 'display'.
Le **Picture_Temporal_Scalable_Extension** contient des informations permettant de supporter la scalabilité temporelle.
Le **Picture_Spatial_Scalable_Extension** contient des informations permettant de supporter la scalibilité spatiale.
Le **Copyright_Extension** indique si le flux est un original ou une copie ainsi que si le flux est protégé par des droits (et dans ce cas fournit le numéro de copyright).

IX.2.2 - Syntaxe MPEG-2 audio



- *En-tête* : structure commune aux 3 couches de MPEG-1 et de MPEG-2.

Le **Syncword** permet au décodeur de se synchroniser sur le début de la frame. Tous les bits de ce champ sont mis à 1.
Le **ID** indique si le PDU est encodé selon MPEG-1 ou MPEG-2.
La **Couche** indique la couche utilisée (1, 2 ou 3).
Le **Bit de protection** indique si la vérification d'erreurs est présente (mis à 0 si oui).
Le **Bit_Rate_Index** indique le bitrate utilisé, indépendamment du mode (stéréo,...).
Le champ **Fréquence d'échantillonnage** indique la fréquence d'échantillonnage utilisée. Si la frame est encodée selon MPEG-1, on aura le choix entre 32, 44.1 ou 48 KHz. Si la frame est encodée selon MPEG-2, on aura le choix entre 16, 22.05, ou 24 KHz.
Le **Bit de bourrage** indique si du bourrage est présent ou non.
Le **Bit privé** n'a pas d'utilisation spécifiée par MPEG.
Le **Mode** indique le mode utilisé (voir section suivante).
Le **Copyright** indique si le flux est protégé par des droits ou pas.
Le **Original/Copy** indique si le flux est un original ou une copie.

- *Vérification d'erreur* :

Ce champ, codé sur 16 bits, est présent si le **bit de protection** dans l'en-tête vaut 0. Il est de type CRC (Cyclic Redundancy Check) et permet de détecter les erreurs dans le flux de bits. Les bits 16 à 31de l'en-tête sont toujours protégés dans les 3 couches.

- *Champ des données audio*:

Pour les couches 2 et 3, la structure du PDU pour ce champ est différente, reflétant une compression et une technique de codage plus complexe.

Le **Bit d'allocation** indique le nombre de bits utilisés pour représenter les échantillons dans les 'sub-band' de chaque canal. Il peut prendre une valeur entre 0 et 15.

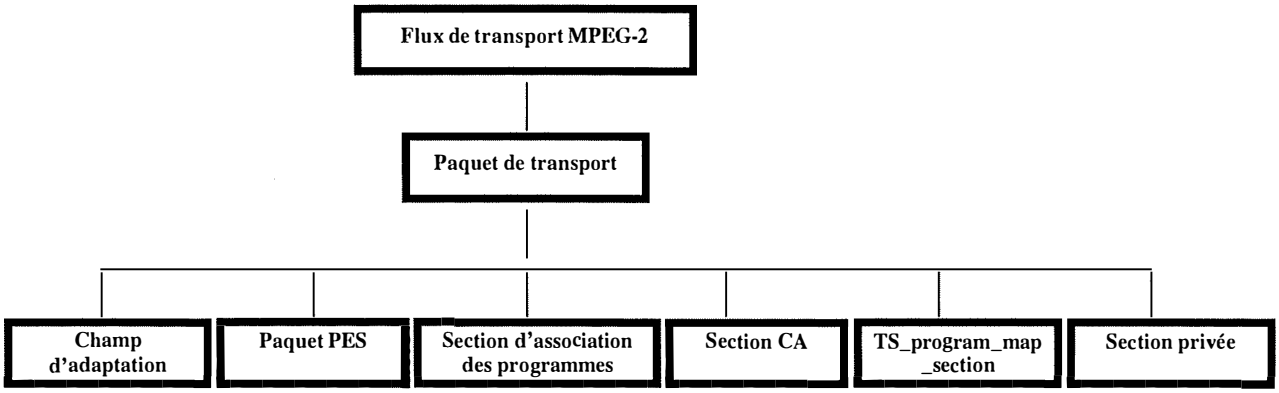
Le **Scale_Factors** indique le facteur par lequel les échantillons doivent être multipliés lors du décodage.

Le **Subband_Samples** contient les échantillons audio.

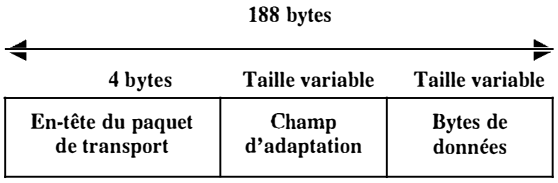
- *Ancillary_Data_Field* :

Le **Ancilary_Bits** est un champ définissable par l'utilisateur. Pour MPEG-2, il permet de transporter les informations d'extension multicanaux.

IX.2.3 - Syntaxe MPEG-2 Système



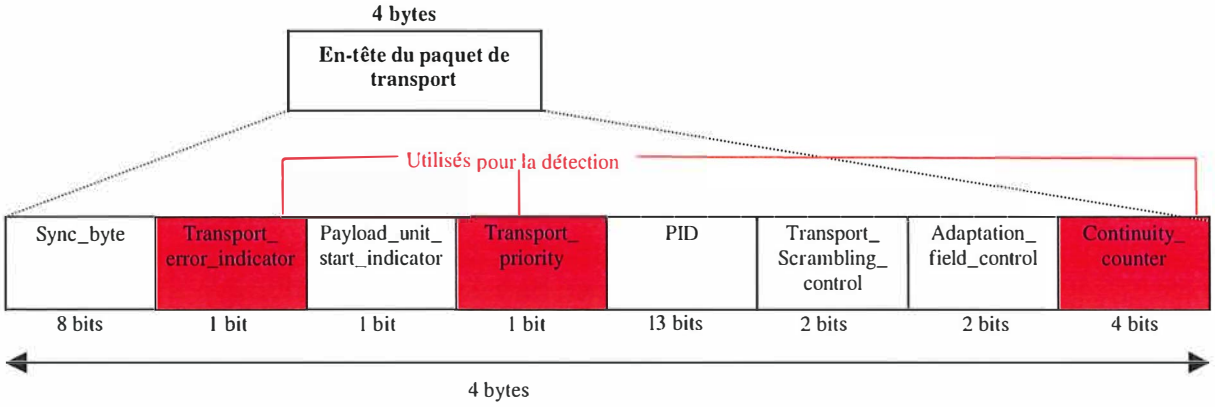
MPEG-2 définit des paquets de transport de taille fixe avec une longueur de 188 bytes. Le paquet de transport MPEG-2 est constitué d'un en-tête de 4 bytes, d'un *Adaptation_Field* de taille variable et d'un *contenu* contenant le paquet PES.



Eléments clefs de la structure des flux de transport :

IX.2.3.1 - En-tête

L'en-tête des paquets du flux de transport fournit des informations qui sont utilisées pour transporter et délivrer les flux. Cela inclut les outils permettant de multiplexer les différents flux d'information.



Le **Sync_byte** a la valeur 57 (en hexadécimal) et est utilisé pour identifier le début d'un paquet de transport.
Le **Transport_Error_Indicator** indique un bit d'erreur dans le paquet de transport.
Le **Payload_Unit_Start_Indicator** indique que le premier byte du *contenu* du paquet de transport est le début d'une *unité de contenu* (par ex, un paquet PES ou une table PSI).
Le champ **Transport_Priority** est utilisé pour indiquer la priorité relative des paquets de transport.

Le **PID** (*Packet IDentifier*) est un des champs les plus importants de l'en-tête. Le PID est utilisé non seulement pour identifier les paquets de transport qui contiennent des données PES provenant du même flux élémentaire, mais aussi pour définir le type de données qui est transporté dans le *contenu* du paquet. Certaines valeurs de PID sont prédéfinies et ont une signification spéciale dans le contexte du standard MPEG-2 Système. Certaines de ces valeurs sont illustrées dans le tableau suivant.

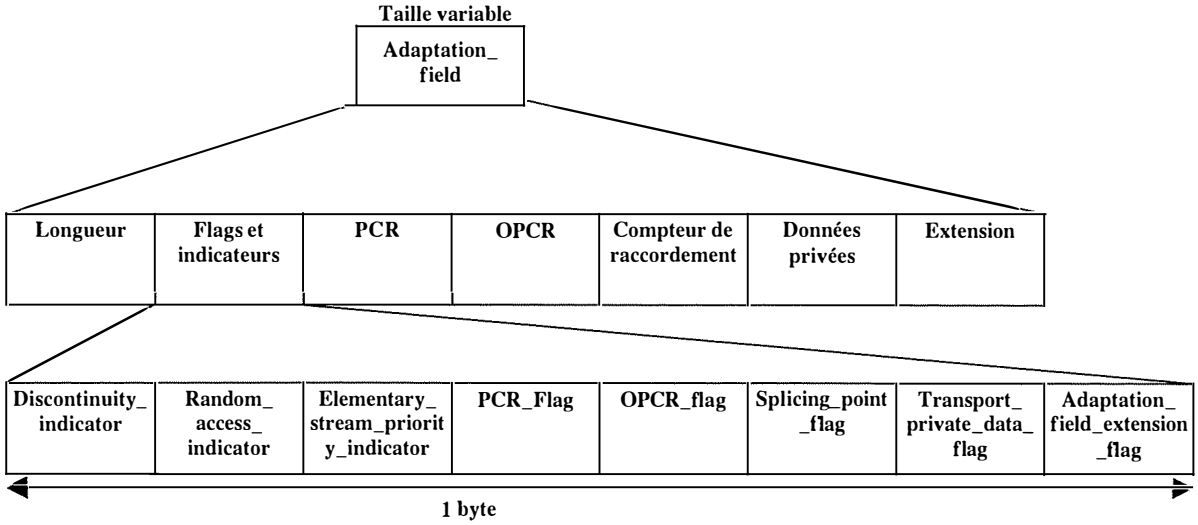
Valeur de PID	Description
0x0000	PAT
0x0001	CAT
0x0002 – 0x000F	Réservé
0x00010 à 0x1FFE	Disponible pour les flux PES, map tables, network tables

Par exemple, pour les paquets de transport qui ont un PID mis à 0, le *contenu* de ces paquets de transport contient une structure particulière appelée *tableau d'association de programmes* (PAT). Les paquets de transport avec un PID mis à 0x10 transportent des données PES en provenance d'un flux élémentaire audio ou vidéo.

Le **Transport_Scrambling_Control** indique le brouillage utilisé pour le *contenu* du paquet.
Le **Adaptation_Field_Control** indique si l'en-tête est suivi par un *Adaptation_field* et/ou un *contenu*.
Le **Continuity_Counter** est un compteur qui est incrémenté à chaque paquet de transport possédant le même PID. Il est remis à 0 lorsqu'il atteint sa valeur maximale. Ce compteur est utilisé pour déterminer la perte de paquets.

IX.2.3.2 - Adaptation_Field

L'*Adaptation_Field* est un champ optionnel dans l'en-tête des paquets de transport qui contient des informations utilisées pour la gestion de l'horloge et pour les fonctions *raccordement* – voir plus loin). Bien que les données qu'il contient soient très importantes pour le traitement des flux de transport MPEG-2, il n'est pas nécessaire dans tous les paquets de transport. De ce fait, ce champ a été déclaré optionnel et est utilisé sur demande dans les paquets de transport.



Le champ Longueur indique la longueur de l'Adaptation_Field.

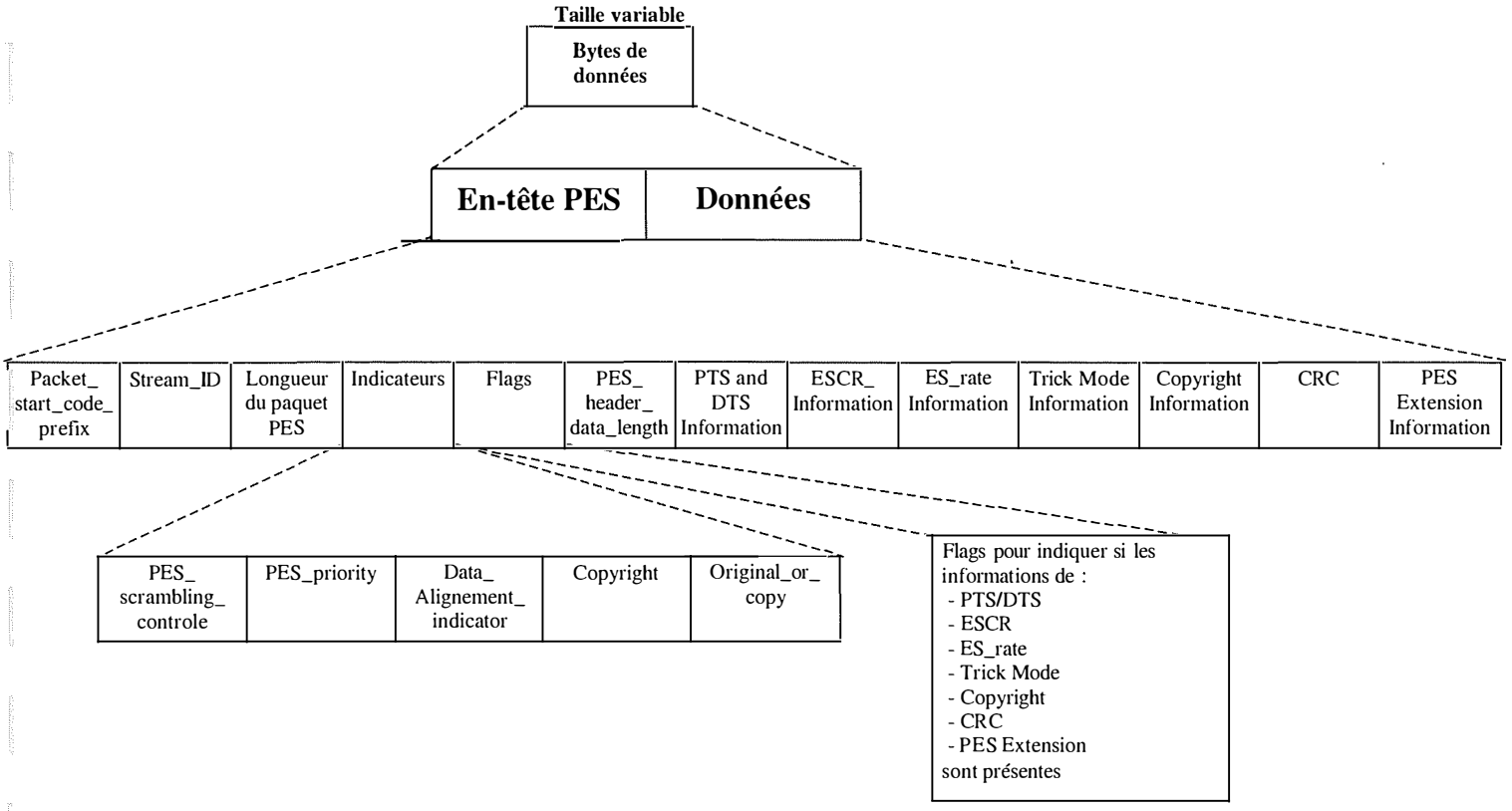
Le champ *PCR* (*Program_Clock_Reference*) est un des champs les plus importants de l'*Adaptation_Field*. Ce champ contient des timestamps qui sont utilisés par le décodeur pour synchroniser son horloge avec celle de l'encodeur.

L'*Adaptation_Field* contient un certain nombre de flags et d'indicateurs au début de la structure.
Les flags déterminent le reste de la structure de l'*Adaptation_Field*. Les bits d'indication, quant à eux, sont utilisés pour donner de l'information au sujet du *contenu*. Par exemple, le bit *Elemntary_Stream_Priority* est activé si le *contenu* contient des données très importantes (comme une I-picture dans le cas de la vidéo).

L'**OPCR_Fields** contient le Program Clock Reference original.
Le **Compteur de raccordement** est utilisé pour les fonctions de raccordement (voir plus bas).
Le champ de **Données privés** est un champ définissable par l'utilisateur.
Le champ **Extension** fournit plus d'information pour supporter le raccordement et le multiplexage des flux de transport.

IX.2.3.3 - Paquet PES

Les paquets PES sont des paquets de taille variable et de format variable.



Packet_Start_Code_Prefix :
Le **Stream_ID** définit le format du paquet PES. Le tableau suivant donne quelques exemples de valeurs de Stream_ID.

Stream_ID	IX.2.3.3.1.1.1.1.1 Description du flux
110x xxxx	Flux MPEG-2 Audio numéro xxxx, contenant des access units audio
1110 yyyy	Flux MPEG-2 Vidéo numéro yyyy, contenant des access units vidéo
1111 0010	Flux MPEG-2 DSM-CC, contenant des données de protocole DSM-CC

Le **PES_Packet_Lenght** indique la longueur du paquet PES.

Les flags, qui peuvent être activés dans l'en-tête des paquets PES, définissent si les informations suivantes sont présentes dans le flux de bits :

- Presentation Timestamp (PTS) et Decode Timestamp (DTS) :

Dans le cas d'un flux audio ou vidéo, les paquets PES peuvent contenir des timestamps pour indiquer quand les données doivent être décodées et présentées. Le DTS est optionnel et est uniquement utilisé si le moment où les 'access units' qui sont décodés diffèrent du moment où ils sont présentés.

- Elementary_Stream_Clock_Reference et Elementary_Stream_Rate :

Champs informationnels qui fournissent un support de temps au décodeur. Si le flux élémentaire n'est pas encapsulé ('embeded') dans un flux de transport, cette information peut être utilisée de façon similaire au champ Clock_Reference de l'*Adaptation_Field* pour les paquets de transport.

- Trick_Mode :

Indique que le *contenu* PES représente un flux (vidéo) spécial. Par exemple, un Trick_Mode_Control_Field peut être mis à une valeur définie comme 'avance rapide', quand une telle fonction est enclenchée dans une application de vidéo à la demande.

- Copyright :

Si ce bit est mis à 1, les données dans le paquet PES sont protégées par des droits. Cette information pourrait être utilisée pour implémenter une protection simple contre la copie. Par exemple, certaines opérations, comme le stockage de données sur disque, pourraient être désactivées après que ce bit ait été vérifié.

- CRC :

Les paquets PES peuvent contenir une valeur de CRC, qui est calculée d'après le *contenu* des paquets PES précédents.

- PES Extension :

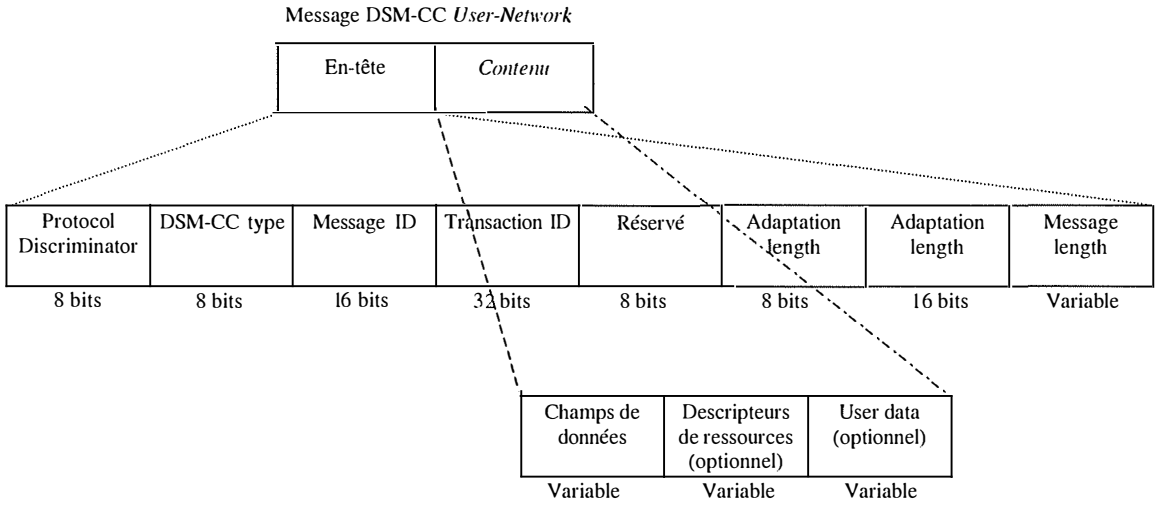
Contient divers champs pour supporter la manipulation de buffers et pour supporter les flux MPEG-1 Système.

Les indicateurs sont utilisés pour donner de l'information additionnelle au sujet du contenu des paquets PES :

- PES_Scrambling_Control : indique si le *contenu* est 'scrambled', et si oui, indique quel 'scrambling' défini par l'utilisateur est utilisé.
- PES_Priorité : permet d'indiquer la priorité du paquet PES
- Data_Alignement_Indicator : indique si le *contenu* débute avec un code de départ audio ou vidéo.
- Copyright : indique si le *contenu* est protégé par des droits
- Original_ou_Copie : indique si le *contenu* est un original ou une copie

IX.2.4 - Syntaxe des Messages User-to-Network

Un message DSM-CC *User-to-network* se compose toujours d'un en-tête et d'un *contenu*.



IX.2.4.1 - En-tête

Le **Protocol Discriminator** identifie ce message comme un message DSM-CC.
Le **Dsmcc_Type** identifie le type de message DSM-CC. DSM-CC distingue deux types de messages pour l'interface *User-to-network*. Il y a tout d'abord les messages de configuration *User-to-network*. Un client utilise ces messages pour se configurer par rapport au réseau auquel il est attaché. Cette configuration peut être initiée par l'utilisateur (messages *UNConfigRequest*, *UNConfigConfirm*) ou par le réseau (messages *UNConfigIndication*, *UNConfigResponse*), ou comme une conséquence d'un utilisateur écoutant un canal broadcast bien connu (*UNConfigIndication*). Après une séquence de messages U-N Config, le client est au courant des paramètres spécifiques du réseau, tels que la façon dont les identificateurs de session sont alloués, ou la façon de communiquer avec le SRM (adresse IP par exemple). La partie U-N Config de DSM-CC est un protocole indépendant. N'importe quelle application qui requiert une configuration initiale avec le réseau peut utiliser cette partie de DSM-CC.

Nom du message	Description
UNConfigRequest	Envoyé de l'utilisateur vers le réseau pour configuration
UNConfigConfirm	Envoyé du réseau vers l'utilisateur en réponse à un UNConfigRequest
UNConfigIndication	Envoyé du réseau à l'utilisateur pour configurer un appareil utilisateur
UNConfigResponse	Envoyé de l'utilisateur au réseau en réponse à un UNConfigIndication

Il y a ensuite les messages de gestion des sessions et des ressources *User-to-network*. Ce groupe de messages est utilisé pour établir et gérer les sessions applicatives vidéo. Il constitue une part essentielle du standard DSM-CC.

Le **messageID** contient 3 parties : le *discriminateur du message*, le *scénario de message* et le *type de message*. Le premier indique si le message est passé entre un client et le réseau ou entre un serveur et le réseau. Le deuxième décrit le scénario dans lequel le message est utilisé (par exemple pendant l'établissement d'une session ou la clôture d'une session). Le troisième indique si l'utilisateur ou le réseau envoie le message et si le message échangé est initié par l'utilisateur ou le réseau. Il peut être un *Request* (utilisateur vers réseau), un *Confirm* (réseau vers utilisateur), ou un *Response* (utilisateur vers réseau en réponse à un *Indication*).

Le **TransactionID** est un identifiant unique pour le traitement du message.
L'**AdaptationLenght** donne la longueur du champ optionnel Adaptation Header.
Le **MessageLenght** donne la longueur du message.
Le **DsmccAdaptationHeader** est un champ optionnel de l'en-tête qui permet l'accès conditionnel ou qui permet à l'utilisateur de définir des informations.

IX.2.4.2 - Contenu

Le format du *contenu* dépend du *messageID* dans le type de message DSM-CC.
Le *contenu* contient des champs de données et des descripteurs de ressource.

IX.3 - MPEG-7

IX.3.1 - Un exemple complet de MediaInformation

```
<MediaInformation>
       <MediaIdentification>
    <Identifier IdOrganization='MPEG'
         IdName='MPEG7ContentSet'> mpeg7_content:news1
    </Identifier>
   </MediaIdentification>

  <MediaProfile>
    <MediaFormat>
         <FileFormat>MPEG-1</FileFormat>
         <System>PAL</System>
         <Medium>CD</Medium>
         <Color>color</Color>
         <Sound>mono</Sound>
         <FileSize>666.478.608</FileSize>
         <Length>00:38</Length>
         <AudioChannels>1</AudioChannels>
    </MediaFormat>
    <MediaCoding>
```

```
            <FrameWidth>352</FrameWidth>
            <FrameHeight>288</FrameHeight>
            <FrameRate>25</FrameRate>
            <CompressionFormat>MPEG-1</CompressionFormat>
        </MediaCoding>
        <MediaInstance>
            <Identifier IdOrganization='MPEG'
                IdName='MPEG7ContentSetCD'>
                mpeg7_17/news1
            </Identifier>
            <Locator>

<MediaURL>file://D:/Mpeg7_17/news1.mpg</MediaURL>
            </Locator>
        </MediaInstance>
      </MediaProfile>
    </MediaInformation>
```

IX.3.2 - *Un exemple complet de DS Creation Information*

```
        <CreationMetaInformation>
            <Creation>
                <Title type="original">
                    <TitleText xml:lang="es"> Telediario
                    (segunda edición)
                     </TitleText>
                     <TitleImage>
                          <MediaURL>
                          file://images/teledario ori.jpg
                          </MediaURL>
                     </TitleImage>
                </Title>

                <Title type="alternative">
                    <TitleText xml:lang="es"> Noticias de la
                    tarde
                    </TitleText>
                    <TitleImage>
                         <MediaURL>
                         file://images/teledario alt.jpg
                         </MediaURL>
                    </TitleImage>
                </Title>

                <Title type="alternative">
                    <TitleText xml:lang="en">Afternoon news
                    </TitleText>
                    <TitleImage>
                         <MediaURL>
                         file://images/teledario en.jpg
```

```
                    </MediaURL>
                </TitleImage>
            </Title>

            <Creator>
                <role>presenter</role>
                <Individual>
                    <GivenName>Ana</GivenName>
                    <LastName>Blanco</LastName>
                </Individual>
            </Creator>
            <CreationDate>1998-06-16</CreationDate>
                <CreationLocation>
                <PlaceName xml:lang="es">Piruli</PlaceName>
                <Country>es</Country>
            <AdministrativeUnit>Madrid</AdministrativeUnit>
                </CreationLocation>
        </Creation>

        <Classification>
            <CountryCode>es</CountryCode>
            <Language>
                <LanguageCode>es</LanguageCode>
                <CountryCode>es</CountryCode>
            </Language>
            <Genre>News</Genre>
            <PackagedType>Information</PackagedType>
            <Purpose>broadcasting</Purpose>
            <AgeClassification>all</AgeClassification>
        </Classification>

        <RelatedMaterial>
            <Master>false</Master>
            <MediaType>Web</MediaType>
            <MediaLocator>
                <MediaURL>www.rtve.es</MediaURL>
            </MediaLocator>
          </RelatedMaterial>
</CreationMetaInformation>
```

IX.3.3 - Autres exemples de D Classification :

```
1) <Classification>
        <MediaReview>
            <Reviewer>
              <Individual>
                <FamilyName> Ebert  </FamilyName>
                <GivenName> Roger </GivenName>
              </Individual>
```

```
            </Reviewer>
            <RatingCriterion>
              <CriterionName> Overall </CriterionName>
              <WorstRating> 1 </WorstRating>
              <BestRating> 10 </BestRating>
            </RatingCriterion>
            <RatingValue> 10 </RatingValue>
                 <FreeTextReview xml:lang="en">
                 Excellent Drama
            </FreeTextReview>
            <FreeTextReview xml:lang="es"> Excelente
            </FreeTextReview>
        </MediaReview>
        </Classification>

2)  <Classification>
        <MediaReview>
          <Reviewer>
             <Organization>
                  <OrganizationName> Blockbuster
                  </OrganizationName>
             </Organization>
          </Reviewer>
          <RatingCriterion>
             <CriterionName>
             Number_of_Rentals_Nationwide
             </CriterionName>
             <WorstRating> 20 </WorstRating>
             <BestRating> 1 </BestRating>
          </RatingCriterion>
          <RatingValue> 14 </RatingValue>
          <FreeTextReview xml:lang="en">
             Top 20 most rented video for the period
             March-April, 2000
          </FreeTextReview>
        </MediaReview>
    <Classification>
```